

RS 232

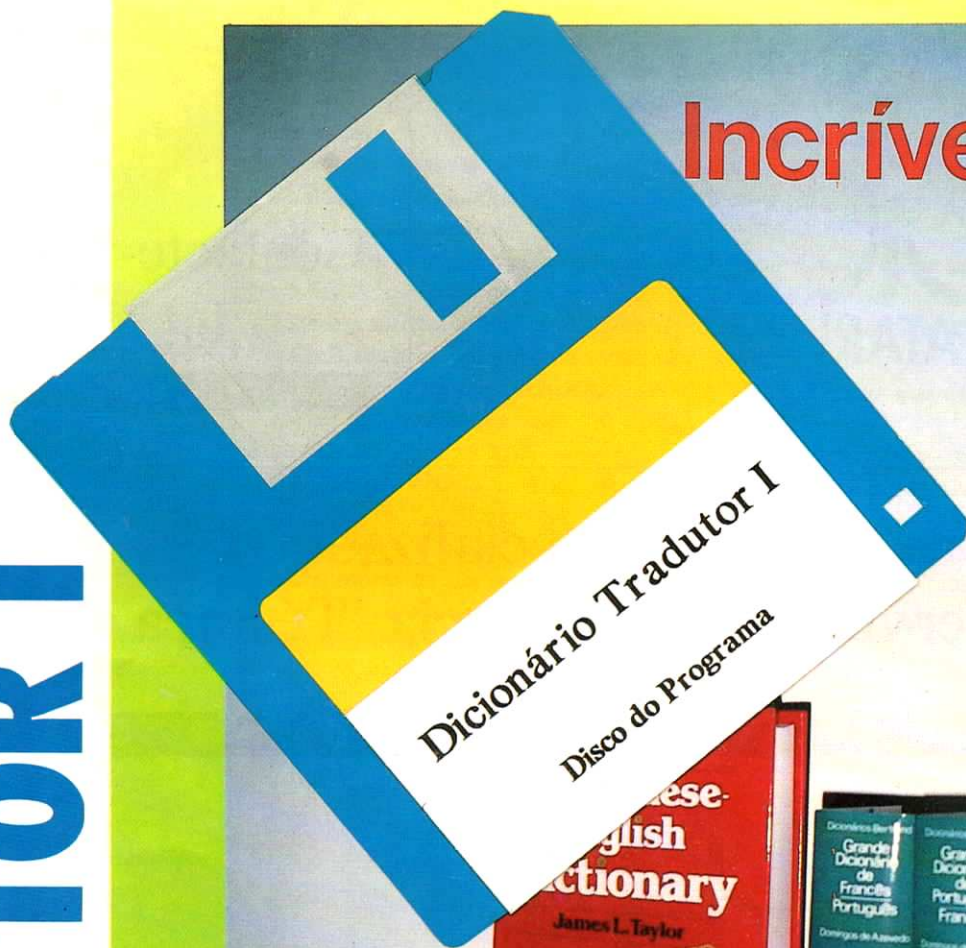
RS 232

INFORMÁTICA

Nº 30 - NOVEMBRO * ANO 3 * PUBLICAÇÃO MENSAL * Preço - 300. Esc.

Software (Dicionário)
TRADUTOR I

Incrível !...



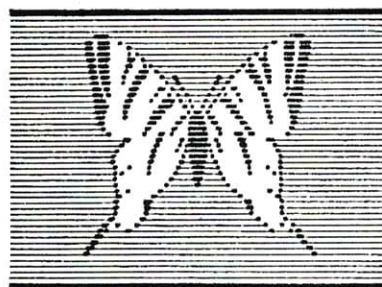


AGORA
TUDO O QUE
PRECISA
PARA O SEU

ATARI

Contacte-nos
Já!

Apoio Especializado
Serviços de Assistência Técnica



MODUS
INFORMÁTICA

ESCRITÓRIO
R. Professor Queiróz Veloso, Lote 1, r/c Dtº
1600 Lisboa
Tel.: (01) - 758 63 60

« RS232-Informática »

Calç. de Palma de Baixo, 15 c/v D
Tel: 726 46 52
1600 Lisboa

PROPRIEDADE

Carlos Aguda

EDITOR

Carlos Aguda

COLABORADORES

Alexandre Rodrigues
Fernando Preces
Pedro Pinto
João Santos
João Fraga
Jorge Gomes
Osvaldo Duarte
Victor Francisco

MONTAGEM/IMPRESSÃO

Gráfica EME SILVA Ldª

DISTRIBUIDORA

MIDESA, S.A.

TIRAGEM

9.000 exemplares

PERIODICIDADE

Mensal

DISTRIBUIÇÃO

Continente
Regiões Autónomas

ASSINATURAS

Continente e Regiões Autónomas
11 números - 3.000\$00
6 números - 1.800\$00
Estrangeiro
11 números - 6.000\$00
6 números - 3.600\$00

« RS232-Informática »
está inscrita na D.G.C.S.
com o N.º 112713

DEPÓSITO LEGAL

N.º20158 / 88

Editorial

Caro Leitor:

Contrariamente ao prometido, fomos obrigados a adiar a nossa edição «Bomba» !!! De facto, quando as coisas não dependem só de nós existe uma maior possibilidade de falha. Sem sermos totalmente culpados, cumpre-nos apresentar-lhe as nossas sinceras desculpas por este “adiamento” que se deve ao facto de não termos previsto qualquer atraso na conclusão dos assuntos que decorrem com empresas estrangeiras, o que realmente veio a acontecer...

Por outro lado, a nossa participação na maior feira de Informática que se realiza entre nós, a INFORPOR, provocou que não tivéssemos tempo para introduzir novas alterações e incluir outros temas de interesse, previstos para este número, bem como dar seguimento ao iniciado “Aprendendo linguagem C”.

Para além de tudo isto, o crescimento de «RS232-Informática» está a ultrapassar as nossas capacidades e impõe-se que dediquemos, de momento, maior atenção ao reforço da nossa estrutura, em prejuízo de outros aspectos da nossa evolução.

As solicitações vindas do exterior, surgiram em número não esperado e quase em simultâneo, encontrando-nos “desprevenidos” para corresponder rápida e eficazmente aos objectivos das mesmas.

Esta é uma situação que nós próprios criámos através do nosso saudável crescimento e, no fundo, sabíamos que estaria próxima, mas não “adivinhámos” a sua dimensão... Estamos satisfeitos por esta “Crise de Crescimento” e ocorre-nos perguntar:

- Onde estão os que, no início, nos apelidaram de “loucos” ?

A «Bomba» está para rebentar, mas não iremos afirmar que acontecerá no próximo mês!!! Não desejamos falhar de novo...

Se o prometemos na edição anterior, foi por não termos previsto quaisquer atrasos e por nos termos deixado arrastar por um excesso de entusiasmo, baseado na certeza dos factos.

Renovando o nosso pedido de desculpas por este «adiamento», acreditando ser merecedores da sua compreensão, iremos cuidar para que a “nossa” «RS232» lhe dê uma agradável surpresa pelo Natal...

Fique connosco!!!

Contamos consigo!!!

Um abraço, de toda a Equipe «RS232».

Sumário

- 4 - NOTÍCIAS
- 8 - TRADUTOR I
- 13 - PROGRAMAÇÃO PC
- 18 - INTRODUÇÃO AO CÓD. MÁQUINA
- 21 - CLUBE DOS LEITORES
- 24 - GESTÃO DA FACTURAÇÃO DE UMA EMPRESA
- 27 - RECURSIVIDADE EM ZX SPECTRUM
- 36 - MSX — Gráficos e Som
- 40 - A MATEMÁTICA NO SEU COMPUTADOR

APRENDENDO LINGUAGEM C - Continua no N.º31
PRAZO DO CONCURSO DE SOFTWARE «RS232»
FOI PROLONGADO ATÉ AO FINAL DE NOVEMBRO.

Os artigos publicados são da exclusiva responsabilidade dos seus autores

EMULADOR OMNIPATH PARA Token Ring.

Este produto está disponível em versões para um utilizador ou em "gateway" permitindo a ligação directa Token Ring via TIC (Token Ring Interface Coopler) ou TRA (Token Ring Adapter) a 4 ou 16 Mbps. Utilizando o protocolo 802.2 DLC Omnipath permite sessões de display e de impressora bem como a transferência de ficheiros quer em configuração stand - alone quer em gateway.

Representados entre nós pela SILICON, a série de produtos Omnipath oferece a possibilidade de utilizar as versões de emuladores da ICOT, ksaver e Ksaver XL, utilizando este último a memória mínima (40 Kb) do PC de importância para os utilizadores de grandes aplicações em micros. As versões Omnipath suportam NetView, o software de gestão da IBM, permitindo ao responsável das comunicações manter ou restabelecer a interligação. Os produtos da ICOT são os primeiros a oferecer ao mercado aquela facilidade, disponibilizando soluções evoluídas de gestão de rede através de gateway .

SITKA Corporation É O NOVO NOME DA TOPS, QUE ANUNCIA ENTRADA NO «MUNDO DOS» EPRI- MEIRO PRODUTO SITKA

Segundo um dos responsáveis desta empresa, "Nós queríamos ter um nome forte, que exprimisse simbolicamente as qualidades da nossa linha de produtos: a família de produtos para redes de dados que possibilita às pessoas que utilizam plataformas diversas - compatíveis MS-DOS, computadores Apple Macintosh e estações de trabalho SUN - comunicar e trabalhar juntos ".

Associado à mudança de nome da companhia, verifica-se um grande dinamismo através de uma nova e mais agressiva estratégia de abordagem do mercado DOS.

Recentemente a SITKA anunciou melhoramentos na versão DOS do software InBox Plus 3.0 Email.

Também foi anunciado por esta empresa a disponibilização do Mail*Link SMTP, um produto que permite a comunicação de dados entre UNIX Mail e InBox Plus, um sistema de correio electrónico originário da TOPS, que "corre" em compatíveis PC e em Apple Macintosh.

Esta ligação dos computadores pessoais com os sistemas UNIX adquire maior relevância numa altura em que os Sistemas Abertos ganham cada vez mais adeptos. A SITKA é representada em Portugal pela H.S.C.Ibérica.

ASANTÉ TECHNOLOGIES COM NOVOS PRODUTOS

Os dois novos produtos desta empresa são as placas de rede Ethernet MacCon+E e o 10T HUB.

O anúncio do 10Base T Hub é a resposta da ASANTÉ à procura que se constata no mercado, de produtos que suportem a especificação Ethernet " 802.3 10Base T Draft 10".

Lançado na MacWorld Expo, em Agosto, este produto associa à tecnologia um preço baixo e uma grande facilidade de instalação. É possível obter a velocidade de 10Mbps numa rede Ethernet, sem os custos e complicações da cablagem coaxial. O Asanté 10T Hub pode ligar até 12 estações de trabalho - utilizando os conectores RJ-45, quer conectores de 50-pin - e pode ser interligado com outras redes de dados. Também recentemente foi anunciada a disponibilização das placas de rede Ethernet MacCon+E para Apple Macintosh SE / 30 e II.

Estas placas beneficiam já de uma arquitectura que permite transferência de dados de 32-bit, o que significa a mais elevada velocidade de transferência de dados existente em placas de rede Ethernet. A H.S.C.Ibérica é a empresa portuguesa que representa a ASANTÉ.

DIGITAL ANUNCIA MARCA REGISTRADA X/OPEN PARA OS SISTEMAS RISC/ULTRIX

A Digital Equipment Corporation anunciou que todos os seus sistemas baseados no sistema operativo ULTRIX passam a exibir a marca de conformidade com as especificações das normas X/Open (XPG3). A Digital está agora autorizada a utilizar a marca registada X/Open nos seus produtos RISC/ULTRIX, bem como na linha VAX/ULTRIX.

Esta conformidade, já anunciada em Junho para a família de sistemas VAX, estende-se agora às famílias DECstation e DECsystem baseadas em processadores RISC, tornando a Digital no construtor com maior base de sistemas conforme as normas X/Open da indústria informática. O sistema operativo ULTRIX da Digital, uma implementação do UNIX baseado na versão de Berkeley, também obedece às normas IEEE POSIX 1003.1/ISO 9945-1 para o interface do sistema operativo, assim como as especificações do X/Open e da Open Software Foundation (OSF).

SOFTWARE PARA TRADUÇÃO COM SOLUÇÃO TOTAL

A JOGOFO Soft lançou recentemente no mercado o Tradutor I. Trata-se de um package de Software totalmente revolucionário em todos os aspectos. É de salientar a inovação que o programa traduz no seu aspecto de apresentação e inovação que ultrapassa em longa escala qualquer Software House Portuguesa, no âmbito deste produto, chegando mesmo, a poder comparar-se com as maiores Software Houses estrangeiras. O Tradutor I utiliza um interface gráfico de janelas único, porque além de nos parecer ser um dos mais simples de utilizar que conhecemos, utiliza gráficos com animação. O mais importante neste programa é sem dúvida a sua solução total de um problema que é a tradução, fazendo-o a 100%.

SOFTLOG NOMEIA A SISMATIC AGENTE NeXT PARA O CENTRO DO PAÍS

Com o alargamento da sua gama de concessões a SISMATIC reforça a sua posição no mercado informático.

Com o NeXT esta empresa vê assim preenchida a faixa de mercado das workstations a qual tem evidenciado um crescimento significativo, tendo registado um enorme acolhimento por parte dos seus clientes. O sistema NeXT, dispõe de uma variada gama de configurações, com a qual responde de forma igualmente satisfatória e eficaz, às exigências de um vasto universo de utilizadores.

Fruto de aprofundados estudos de desenvolvimento, o sistema NeXT tem na inovação tecnológica a definição das suas maiores potencialidades.

1- Inovação tecnológica:

- a) Arquitectura optimizada que concorre directamente para a qualidade do produto final.
- b) Tecnologia pioneira no arquivo e acesso à informação de forma extremamente fiável.
- c) Qualidade de som C.D., permitindo que este seja integrado com as aplicações do dia a dia.
- d) Sistema unificado de imagem (Display PostScript) de forma que esta seja idêntica quer no ecrã quer na impressora.
- e) Interface intuitivo com acesso às capacidades de Networking e Multitask do sistema operativo Unix.
- f) Correio electrónico multimédia, combinando texto, gráficos e voz.
- g) Um novo sistema de desenvolvimento reduzindo o tempo necessário na produção de software.

2- Software:

Respondendo às diferentes solicitações do mercado, existem já inúmeros packages disponíveis nas áreas de Gestão, Bases de Dados, Desktop Publishing, etc., onde se destacam nomes como Informix, Ingres, Oracle, Lotus e Ashton Tate.

A Apple ADERE AOS CONSÓRCIOS UNIX INTERNACIONAL E OPEN SOFTWARE FOUNDATION

Esta decisão acelera a anuência da Apple às normas industriais, assegurando deste modo o seu acesso antecipado às tecnologias UNIX.

Adicionalmente, a Apple garantiu a harmonização da sua plataforma UNIX Macintosh, o A/UX, com o X/Open Portability Guide (XPG).

Segundo Ron Lang, gestor de Marketing do A/UX, a Apple oferece aos seus utilizadores ferramentas poderosas e únicas - as vantagens do popular ambiente Macintosh combinados com os standards industriais.

Ao aderir a ambos os consórcios a Apple poderá avaliar toda a tecnologia apresentada e adoptar a que melhor corresponde às necessidades dos seus clientes. A aderência ao XPG facultará mais uma norma importante aos seus clientes nas empresas, universidades e administração em todo o mundo.

O A/UX 2.0 oferece ambiente Macintosh aos utilizadores que necessitem de UNIX. Apresenta o familiar e intuitivo ambiente de trabalho Macintosh com todas as aplicações que fizeram o sucesso destes computadores. O A/UX 2.0 permite que o utilizador disponha de um único sistema na sua secretária para o trabalho diário, incluindo aplicações Macintosh, UNIX e X Windows System, e ainda programas em MS DOS e Motif utilizando produtos de fabricantes independentes.

Interlog RENOVA CONTRATO COM A APPLE COMPUTER.

A Interlog renovou o seu contrato com a Apple Computer para representação dos produtos Apple Macintosh em Portugal, garantindo igualmente essa representação para Angola e Moçambique.

A celebração do contrato, com duração de três anos, teve lugar em Paris a 21 de Setembro.

TOPICAL SOFTWARE

Esta empresa, formada em Outubro de 1988 (data em que começámos «RS232»), tem a sua sede na COVILHÃ e insere-se no programa de Iniciativas Locais de Emprego do Instituto do Emprego e Formação Profissional.

A produção de Software, a prestação de Serviços de Consultadoria e Assistência, venda de Hardware e Consumíveis, são, em conjunto com a Formação Profissional, as principais áreas de actuação desta empresa. A programação, vocacionada para suprir as enormes lacunas existentes no sector Têxtil de Tecelagem e Confecções, abrange também outras áreas como sejam a da gestão de Ficheiros Médicos, de Colectividades e Associações, etc. Para o desenvolvimento de software de elevada qualidade a Tropical Software utiliza modernas ferramentas de programação em microcomputadores como o Turbo Pascal versão 5.5 (da Borland Internacional).

A Formação Profissional ocupa grande parte da sua actuação, sendo o próprio Instituto do Emprego e Formação Profissional o seu melhor cliente, com mais de 600 horas de formação prestadas em 1990, assim como outras instituições.

O departamento comercial, que executa a venda de Hardware, Software e Consumíveis, procura neste momento «Business-Partners» por todo o país, pretendendo criar uma rede de distribuidores do Software-Tropical Software.

CITIZEN, HSC, INFOLOGIA, INFORGAL, LAUDAT, SCHNEIDER, TDK, UNISYS, VISEU INFORSOFT, são algumas das companhias e marcas representadas pela Tropical Software.

Para além de toda a sua actividade ligada ao sector Informático, esta empresa tem apoiado diversas iniciativas desportivas realizadas na região e compõem e copiam gratuitamente dois jornais de iniciativa Juvenil. «RS232» felicita-vos e coloca-se à vossa disposição para vos apoiar no que entenderem útil. (Mega BYTE?)

DIGITAL IMPLEMENTA REDE EARN

A Digital Equipment Corporation desenvolveu recentemente um programa para as aplicações e protocolos da Rede Europeia de Investigação Académica (EARN) que obedece às normas públicas de Interligação de Sistemas Abertos (OSI). Graças a este novo suporte, os utilizadores da mais importante rede académica de investigação da Europa, vão poder comunicar com sistemas informáticos de diferentes construtores.

A EARN, primeira rede informática que liga universidades e instituições de investigação da Europa, Médio Oriente e África, solicitou esta colaboração da Digital com o objectivo de libertar a rede das restrições da rede proprietária IBM e ao mesmo tempo transferir a rede para as normas públicas de Interligação de Sistemas Abertos OSI (Open Systems Interconnect) estabelecidas pela ISO (International Standards Organization).

A ligação dos utilizadores da EARN, que já ultrapassou os 70 mil através de cerca de 700 computadores em 450 universidades e centros de investigação em 27 países, é efectuado através de linhas de telecomunicações. Por seu lado, os utilizadores desta rede podem transferir ficheiros de todos os tipos, desde dados, programas ou documentos, receber ou enviar mensagens electrónicas de ou para um ou mais utilizadores, trocar mensagens interactivas em modo de diálogo, partilhar os recursos dos computadores e ter acesso a aplicações, bases de dados e a bibliotecas remotas.

Este suporte desenvolvido pela Digital é considerado como a primeira rede internacional que observa as normas ISO / OSI até à Fase 5. Uma outra característica atraente é o facto de utilizar produtos normalizados de outros construtores. Os pacotes X.25 são DPN-100 da Northern Telecom, o software OSI é constituído por produtos por produtos normalizados da Digital e o software de Entrada de Tra-

balhos na Rede é o JNET, da Joiner Associates. A maior parte do tráfego de informações passa agora a efectuar-se através do suporte para o qual a Digital garante um tempo maior de utilização, o que por outro lado proporciona aos utilizadores do EARN um maior grau de confiança e uma transferência mais rápida de informações.

No entanto, com este suporte, os utilizadores podem continuar a utilizar os protocolos originais IBM bi-síncronos RSCS sem notar qualquer diferença quando a transferência for efectuada.

Os sistemas VAXserver da Digital gerem a conversão de protocolos entre as camadas inferiores de protocolos NJE/BSC, NJE/SHA ou NJE/TRCI e as camadas inferiores de protocolos do sistema ISO/OSI, o que significa que é mantido um serviço contínuo de Entradas na Rede através da camada Sessão/Transporte do OSI.

IBM ANUNCIA CIM PARA PEQUENAS E MÉDIAS EMPRESAS

Esta nova oferta, "CIM Series/400" (para empresas industriais) possibilita a partilha de dados entre vários departamentos - desde a concepção e engenharia, à expedição dos produtos para o Cliente - e permite a integração da informação através da ligação de vários computadores e aplicações.

Os dados são introduzidos no Sistema apenas uma vez, estando disponíveis para qualquer tipo de operação quer se trate de encomendas de matéria prima, da última versão de um desenho ou do código de prioridade de uma remessa.

O "CIM Series/400", integrará também várias plataformas de hardware IBM, que desempenharão funções especializadas. O AS/400 ocupará-se do planeamento da produção (MAPICS/DB) e de outras aplicações de gestão; o RISC System/6000 processará as aplicações de engenharia tais como CAD/CAM e o PS/2 será

o principal interveniente na gestão das operações fabris.

Através de ferramentas especializadas de integração, é também possível ligar os programas de CAD - CATIA ou CADAM - no RISC System/6000 - à base de dados de gestão do AS/400. O mesmo se aplica ao IBM C.A.D., que corre em PS/2. Deste modo, os engenheiros e os projectistas podem gerar ou actualizar desenhos com maior eficiência, através do acesso à base de dados do AS/400.

Além disso, os dados do CAD ficarão disponíveis em qualquer ponto da Empresa, facilitando a tomada de decisões de produção ou de negócios.

Assim se concretiza o anúncio de Outubro, através do qual a IBM definiu a Arquitectura CIM e declarou a intenção de proporcionar a implantação dessa arquitectura nos Sistemas AS/400.

A UNITRON JÁ ESTÁ EM PORTUGAL...

Soubemos esta informação em meados de Outubro e tivemos oportunidade de ver alguns dos muitos produtos que esta prestigiada empresa possui, mas pediram-nos que guardassemos segredo por mais algum tempo, no que se refere a QUEM vai comercializar os referidos produtos em Portugal.

A preços extremamente competitivos, os donos do «segredo» esperam impor-se no nosso mercado, com a certeza de que o binómio preço-qualidade lhe facilitará a tarefa.

Por nos terem pedido segredo, até porque ainda há muita coisa para concretizar e preparar em termos promocionais, não podemos adiantar muito mais, embora a tentação seja forte...

Vamos cumprir a nossa promessa e esperamos, no próximo número, apresentar uma desenvolvida divulgação dos produtos da UNITRON e desvendar quem os passará a representar entre nós. Se o não fizermos, é por não haver luz verde...

COMPAQ SYSTEMPRO ATINGE PERFORMANCE RECORD

A Compaq Computer Corporation revelou os resultados auditados de um teste que demonstra que o COMPAQ SYSTEMPRO corre o programa Oracle Server sob o SCO Unix com o SCO MPX da Santa Cruz Operation oferecendo uma performance de processamento transaccional superior à dos minicomputadores e mainframes.

O teste TPI para servidores de bases de dados é geralmente reconhecido como critério standard para avaliar a capacidade de um computador num ambiente de processamento transaccional "on-line" em bases de dados.

Com dois processadores 486/33 instalados, o COMPAQ SYSTEMPRO realizou mais de 42 transacções por segundo.

Com esta combinação, os utilizadores conseguem atingir a performance que normalmente se espera de um minicomputador topo da gama, e obtêm um custo por transacção inferior ao que qualquer outro computador (incluindo PC's, minicomputadores e computadores de grande porte) hoje pode oferecer.

Esta espectacular performance é o resultado de tecnologias inovadoras, como as drive arrays, capacidade de multiprocessamento, e a memória ServerCache de 512 Kbytes incorporada nos modelos baseados no 486/33 do COMPAQ SYSTEMPRO. Os testes TPI foram auditados por TOM SAWYER, consultor da Codd and Date, e foram divulgados num relatório intitulado "Relatório dos Auditores sobre os resultados do teste TPI para UNIX Server - ORACLE e COMPAQ SYSTEMPRO".

Este equipamento foi testado com várias combinações dos processadores 386/33 e 486/33 correndo o ORACLE Server para a versão 6.0.27 do SCO UNIX/MPX e versão 3.2.2 do SCO UNIX System V/386 com a versão 1.1 do MPX, que estará disponível no último trimestre deste ano.

A configuração com dois processadores 486 resultou em 42.4 transacções por segundo (tps), com um único 486 atingiu 33.4 tps, com dois 386 apenas 22.5 tps, e um processador 386 obteve uma performance de 16.9 tps.

Exemplos de ambientes de bases de dados que requerem elevada capacidade de processamento transaccional incluem a actividade bancária, sistemas de reservas e sistemas de entrada de encomendas. Segundo um dos responsáveis para a Divisão UNIX da Oracle Corporation, "o Oracle oferece a elevada capacidade que as aplicações para processamento transaccional exigem, graças à sua arquitectura cliente/multiservidor. O ORACLE foi concebido para uma utilização eficiente de ambos os processadores no Compaq Systempro; os utilizadores podem consultar

a base de dados em paralelo, armazenar na memória em paralelo, e executar transacções em paralelo.

Isto permite obter a performance óptima quando se utilizam múltiplos processadores no COMPAQ SYSTEMPRO".

Lançado em Novembro de 1989, o Compaq Systempro está vocacionado para responder a um vasto leque de requisitos de servidores de rede e multiutilizadores - desde redes de 15 utilizadores para partilha de recursos até aplicações complexas de bases de dados departamentais. Os modelos baseados no processador 386/33 estão disponíveis em todos os concessionários Compaq, assim como os baseados no 486/33.

A Oracle também anunciou vários Servers para ambientes PC UNIX, para além do ORACLE para o ambiente SCO UNIX/MPX, que já se encontra disponível, em versão multiutilizador e versão servidor.

Os utilizadores vão poder ligar uma variedade de equipamentos - terminais, PCs, computadores Macintosh e estações UNIX - a um COMPAQ SYSTEMPRO correndo ORACLE.

Entre nós, como é do conhecimento do leitor mais atento, os computadores COMPAQ são representados pela DÉCADA.

DÉCADA

COMPAQ

R. PEDRO NUNES, 47 C 1000 LISBOA

Já divulgado e publicitado em edição anterior, este Software merece de novo a nossa atenção em virtude do interesse que tem suscitado entre os nossos leitores e empresas que dele tiveram conhecimento.

de facto, só recentemente tivemos possibilidade de "ver" este software e compreender melhor a dimensão das suas capacidades.

A maior parte das pessoas gostariam de falar e escrever correctamente uma língua estrangeira, outras necessitam urgentemente de escrever nessa língua por questões de trabalho e há pessoas que ainda andam a aprender essa língua.

Para todas essas pessoas, o TRADUTOR I (apesar de não falar nenhuma língua) traduz textos correctamente, em todas elas. Este Software foi elaborado num sistema que consegue traduzir quase sem margem de erro (0,5%), isto porque obedece a 100% às técnicas de tradução de cada país.

Desta maneira, este package consegue traduzir qualquer texto correctamente mesmo que este contenha frases idiomáticas e técnicas de tradução complicadas.

Por tudo isto, o TRADUTOR I é uma ferramenta indispensável tanto para o estudante como para o mais sofisticado tradutor de línguas, sendo o sistema mais completo (a nível de tradução) que existe no mercado.

Este programa foi dividido em três níveis, sendo o nível três o mais baixo, isto para que fosse acessível a todos os utilizadores. Para além do que já foi dito em edição anterior, passamos a especificar o que cada nível contém:

NÍVEL 1 :

Dicionário Manager
Criador de novos Dicionários
Alterador de dicionários
Tratamento e tradução de textos
Processador de texto
Utilitários CRIA-DIC
Utilitários JUNTA-DIC
Dicionário de Português - Inglês
Manuais

NÍVEL 2:

Dicionário Manager
Alterador de Dicionários
Tratamento e tradução de textos
Processador de texto
Dicionário de Português - Inglês
Manuais

NÍVEL 3:

Dicionário Manager
Dicionário de Português - Inglês
Manuais

Os dicionários de Inglês - Português, em fase de acabamento, serão adquiridos separadamente, ficando o seu custo nos seguintes valores:

Com CodePage = 10.000\$00+ IVA
Sem CodePage= 8.000\$00 + IVA

Complementando esta informação, voltamos a mencionar os valores referentes a cada um dos níveis:

Nível 1 = 119.000\$00+IVA
Nível 2= 52.000\$00+IVA
Nível 3= 25.000\$00+IVA

Versão Demo = 2.500\$00+IVA

Nesta nova abordagem sobre o TRADUTOR I, informamos os nossos leitores que este software pode ser adquirido por intermédio da revista, devendo para o efeito enviar cheque ou vale postal de valor equivalente ao nível pretendido, acrescido de 300\$00 para portes, como é feito para outros produtos.

PRINCIPAIS CARACTERÍSTICAS DO PROGRAMA:

- 1 - Possibilidade de qualquer utilizador criar novos Dicionários, independentemente dos que já possam existir. (nível 1)
- 2 - Possibilidade de acrescentar, alterar ou eliminar, tanto palavras como Técnicas de Tradução nos Dicionários já existentes no mercado. (níveis 1 e 2)
- 3 - Tradução sem margem de erro, com todas as Técnicas de Tradução de cada país. Correção de ortografia dos textos, procura de palavras por aproximação, etc. (níveis 1, 2 e 3)
- 4 - Recuperação automática de Dicionários (Palavras e Técnicas de Tradução) e de todos os ficheiros com que o programa trabalha, mantendo assim a impossibilidade de ocorrerem erros a nível de ficheiros e Dicionários. (níveis 1, 2 e 3)
- 5 - Tradução de palavras por aproximação e com toda a informação que contém um Dicionário. (níveis 1, 2 e 3)
- 6 - Possibilidade de traduzir da palavra original para a tradução e vice-versa. (níveis 1, 2 e 3)
- 7 - Possibilidade de tradução por chaves programadas e chaves gramaticais. (níveis 1, 2 e 3)
- 8 - Importação de textos de Processadores mais conhecidos (Word, DW4, Write, WA, etc,), desde que estejam gravados em

forma ASCII. Tradução automática desses textos. (níveis 1 e 2)

9 - Exportação de textos traduzidos em forma de ASCII para qualquer Processador. (níveis 1, 2 e 3)

10 - Possibilidade de manter a formatação do texto original ou alterar a formatação num menu de configuração. (níveis 1 e 2)

11 - Configuração do programa definida pelo utilizador. (níveis 1 e 2)

12 - Processador de texto próprio. Algumas características:
Word-wrap, Browse, Tabulações, Seguranças automáticas, Formatação de página, correcção de ortografia, etc. (níveis 1 e 2)

13 - Interface de Janelas Gráfico com animação, podendo correr com o Windows. (níveis 1, 2 e 3)

14 - Impressão das Traduções em qualquer impressora (desde agulhas a Laser). (níveis 1, 2 e 3)

15 - Impressão ordenada de palavras do Dicionário com a respectiva tradução e informação, tendo a hipótese de escolher a chave (língua, tradução ou chave programada) e a respectiva listagem da palavra "X" à palavra "Y". (níveis 1, 2 e 3)

16 - Sistema de ajuda bastante completo que ensina desde a forma de utilização dos menus até à resolução de erros.

17 - Rotina de erros do utilizador ON LINE com gráficos e animação demonstrando os procedimentos para a resolução do erro.

REQUISITOS MÍNIMOS:

512K de RAM, Disco Duro de 20Mb, Placa Gráfica (Hercules, CGA, EGA ou VGA).

DICIONÁRIOS DISPONÍVEIS:

Refª INGL_A1.IBM (versão teclado português IBM PS/2 e compatíveis)

Português - Inglês -> ACM : 171.000 Palavras
9400 Técnicas de Tradução

INGL_A1.PC (versão teclado multi-país PC e compatíveis)

Português - Inglês -> ACM: 171.000 Palavras
9400 Técnicas de Tradução.

Outros dicionários em fase de acabamento:

INGL_PORTUGUÊS ; PORTUGUÊS-FRANCÊS -
FRANCÊS-PORTUGUÊS; PORTUGUÊS-ALEMÃO e
ALEMÃO-PORTUGUÊS.

O prazo de entrega dos dicionários já disponíveis é de 8 dias úteis após recepção de pedido.

A disponibilidade dos restantes dicionários será divulgada em próximas edições.

PEDIDOS A:

«RS232 - INFORMÁTICA»

Calç. de Palma de Baixo, 15 c/v Drtª

Tel.: (01) - 726 46 52

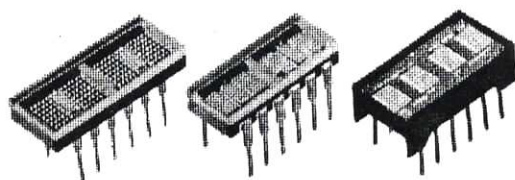
1600 LISBOA

Distribuidor
Exclusivo
para

LISBOA / arredores
e Zona SUL do País

ACEITAM-SE AGENTES

Encomende JÁ o seu TRADUTOR I !!!
O fim dos seus problemas em traduzir textos
ou convertê-los para uma língua estrangeira.



MICROELECTRÓNICA

- Implicações técnicas e económicas

Este foi o tema de uma conferência organizada pela SIEMENS, em Outubro, na qual foram abordadas tanto questões técnicas como económicas decorrentes da evolução no domínio da microelectrónica.

Esta conferência teve como orador o Dr. F.S.Becker, da Siemens AG, um dos responsáveis pelo trabalho de investigação para o desenvolvimento do chip de 4 Mega bits, efectuado nos laboratórios desta empresa em Munique e, pelo interesse da mesma,, deixamos aos nossos leitores o que nela foi dito.

Microelectrónica - uma ideia revolucionaria o mundo

Poucas invenções deixaram um cunho tão marcante na vida humana como a microelectrónica. Passatempos, processamento de dados no escritório, automação industrial e recolha de dados para a defesa do meio-ambiente seriam impensáveis sem a ajuda dos mini-cérebros cristalinos. A rápida ascensão da microelectrónica deve-se a uma simbiose única entre a ciência e a economia. Para se compreender esta situação é necessário considerar tanto os aspectos técnicos como os económicos.

Há três décadas atrás, a invenção do circuito integrado iniciou uma revolução técnica que ainda não acabou.

A soldadura morosa de componentes, como transistores, diodos ou condensadores, foi substituída pela sua produção em simultâneo num "chip" de silício. Dez anos mais tarde era já possível integrar 1000 transistores num só "chip". Hoje em dia, o mercado oferece módulos com quase 10 milhões de componentes activos e o "gigabit", com um bilião de transistores integrados, já parece possível.

Vários factores contribuíram para o aumento da capacidade dos "chips". Estruturas cada vez mais finas são integradas em superfícies de "chips" cada vez maiores. Desde há alguns anos recorre-se também activamente à terceira dimensão.

A delicadeza real das estruturas fica patente através da comparação com um cabelo humano, o qual é 50 vezes mais expesso do que elas.

Os módulos de memória desempenham um papel especial na microelectrónica, porque são produzidos aos milhões e apresentam uma estrutura muito regular.

Graças a esta propriedade, eles são um veículo de teste ideal para optimizações baseadas em análises estatísticas de novas técnicas de fabrico. Por isso as memórias constituem tradicionalmente a "ponta de lança" do progresso tecnológico da microelectrónica.

Alguns anos depois, o processo de fabrico assim ensaiado é utilizado para a produção de outros circuitos integrados, por exemplo os microprocessadores.

Os módulos de memória não são concorrentes mas sim complementos dos suportes magnéticos, visto que desempenham o papel de "memórias voláteis" na técnica digital.

Consoante a aplicação exigisse maior velocidade ou menor consumo de energia, assim foram desenvolvidos vários princípios para armazenagem de informações.

No caso das SRAM - Static Random Access Memory - , é a posição do interruptor flip-flop que define o bit individual, ou seja, a informação básica "sim / não".

A DRAM - Dynamic Random Access Memory - , pelo contrário, lê a informação através de um transistor selectivo para um condensador e estabelece assim a informação 0/1.

Este tipo de memória chama-se "dinâmico", por causa da carga do condensador diminuir gradualmente em consequência de correntes de fuga, exigindo a regeneração periódica de um circuito lógico integrado num "chip".

Para o fabrico de uma DRAM de 4 Mbits, ou seja, de uma memória capaz de armazenar exactamente 4.194.304 bits, são necessárias cerca de 450 fases de processo, repetidas periodicamente em grupo. Em todas estas fases é exigido um elevado grau de perfeição, superior a 99%. Com "apenas" 99% e após 450 fases a 99^{450} , o aproveitamento final de "chips" operacionais diminui para menos de 1%.

Uma comparação facilita a compreensão do problema: se um "chip" de 4 Mbits da SIEMENS, com dimensões de 14 x 6,5mm², fosse aumentando pelo factor 10.000, tingindo o tamanho de um campo de futebol, teria espaço par 270 Km de condutores de alumínio. A camada de isolamento mais fina teria 0.13mm de espessura e não toleraria mais do que uma partícula de sujidade do tamanho da cabeça de um alfinete!



CAF[®]

MICROCOMPUTADORES

**UMA SOLUÇÃO À MEDIDA
DAS SUAS NECESSIDADES**

PORTÁTEIS XT / AT

8088 / 10 Mhz

80286 / 16 Mhz

80286 / 20 Mhz

80386 / SX

80386 / 20 Mhz

80386 / 25 Mhz

80486 / 25 Mhz

DISTRIBUIDOR OFICIAL

**TEMOS A SOLUÇÃO
PARA O SEU PROBLEMA**

- * OPEN ACCESS II
- * CLUBES DE VIDEO
- * MEDIADORES DE SEGUROS
- * MEDIADORES IMOBILIÁRIOS
- * CONTROLO DE VENDEDORES
- * ODONTOLOGISTAS
- * OPTICOS
- * GESTÃO DE LOJAS (Pronto a vestir / Sapatarias)
- * CENTROS DE RADIOLOGIA
- * GESTÃO AGRO PECUÁRIA
- * ADVOGADOS
- * AGÊNCIAS DE DOCUMENTAÇÃO
- * GESTÃO COMERCIAL
- * CONTABILIDADE / SALÁRIOS

- PRODUTORES DE SOFTWARE -

TSI - Tecnologias e Soluções Informáticas

R. Morais Soares, Nº 126 - 3º Esq. 1900 Lisboa Tel: 01-525202

O aumento das superfícies e estruturas ainda mais delicadas tornarão os futuros "chips" ainda mais susceptíveis a defeitos. No caso das DRAMS, desenvolveu-se no passado, em cada 3 anos, uma nova geração com 4 vezes mais capacidade de memória. Depois das DRAMS de 4 Mbits, introduzidas no mercado em 1989, seguir-se-ão as de 16 Mbits em 1992 e as de 64 Mbits em 1995.

Depois disso será mais difícil prever o desenvolvimento, dado que a geração das estruturas exigirá provavelmente o recurso a raios X e os aspectos de rentabilidade económica ainda não estão esclarecidos.

Actualmente, os custos de desenvolvimento aumentam 50% em cada geração de memórias e os custos de fabrico de 50% a 100%. O número de empresas capazes de financiar tais verbas é cada vez menor e os acordos de cooperação e as fusões aumentam.

A microelectrónica como factor económico

A microelectrónica é, hoje em dia, um produto de massas, cujas vendas mundiais em 1988 ascenderam a 110 biliões de marcos, dos quais 2,8 biliões de marcos na R.F.A..

Contudo, estes números não reflectem totalmente a importância dos "chips", indispensáveis para muitos produtos modernos. Só na R.F.A. foram produzidos em 1988 bens no valor de 600 biliões de marcos, desde maquinaria, electrónica, mecânica de precisão, óptica, relojoaria, indústria automóvel, equipamento de escritório e de processamento de dados.

Neste conjunto de produtos, o valor económico dos "chips" tem um papel subordinado, mas em termos de capacidade eles são decisivos. Sem a inteligência artificial dos "chips", os produtos de um país altamente industrializado não podem concorrer no mercado internacional. Sendo certo que uma parte dos "chips" pode ser importada de outros países, é igualmente certo que um país não pode dispensar a longo prazo uma base própria de microelectrónica - os circuitos integrados de aplicações específicas exigem um nível cada vez mais elevado de conhecimentos, que terão de ser adquiridos a outrem no caso de importação dos "chips". Além disto, o factor tempo desempenha também um papel fundamental, tendo em conta que dentro das grandes empresas japonesas os protótipos de novos "chips" são disponibilizados primeiro para a própria empresa antes de passarem para a concorrência europeia.

O primeiro a montar os circuitos integrados mais modernos nos seus produtos garante um avanço muitas vezes decisivo em relação aos concorrentes.

Sem uma indústria de microelectrónica própria, é quase impossível criar e manter a base industrial indispensável às instalações altamente complexas para a produção de "chips".

Face aos efeitos dificilmente previsíveis que a alteração de uma fase do processo de produção de "chips" pode ter sobre a funcionalidade do circuito, devem ser desenvolvidas novas instalações em estreita cooperação com o fabricante dos "chips". É óbvio que o "Know-how" essencial só estará mais tarde à disposição de terceiros...

Em nenhuma outra indústria o factor tempo é tão decisivo como na microelectrónica; o rápido progresso técnico condiciona uma rápida queda de preços. Dificilmente existirá outro produto de massas cujo fabricante se tenha de contentar com apenas um décimo do lucro inicial, após poucos anos de fabrico.

A deterioração dos preços leva por sua vez as empresas a lançar componentes mais integrados, a fim de melhorar a sua margem de lucro.

Há 20 anos, como inventores do circuito integrado, os E.U.A. dominavam o mercado internacional. Nos últimos anos, porém, o Japão tornou-se um concorrente de respeito, com uma política selectiva e agressiva, conquistando alguns produtos, por exemplo memórias dinâmicas, quase 90% do mercado mundial. Embora os E.U.A. ainda detenham uma posição forte nos microprocessadores e equipamentos de produção de "chips", os japoneses já estão a avançar em força.

A Europa desempenha sobretudo um papel de consumidor, importando 2/3 das suas necessidades de "chips".

Devido aos efeitos negativos já aqui esboçados, os europeus empenharam-se nos últimos anos em diversos projectos, como por exemplo o projecto Mega - a DRAM de 4 Mbits da SIEMENS - lançado no mercado em 1989.

A iniciativa JESSI (Joint European Submicrom Silicon) é a sucessora deste projecto e visa um reforço da investigação e da indústria de equipamentos.

Mais de uma centena de empresas europeias querem cooperar e investir mais de 8 biliões de marcos nesta iniciativa. Mesmo assim serão necessários esforços adicionais para garantir à Europa uma posição relevante na tecnologia do futuro - a microelectrónica.

Sem isto, a Europa corre o perigo de perder a "guerra dos chips" e, consequentemente, a sua posição no quadro das potências industriais.

Esta é, pois, uma panorâmica da situação actual em relação à evolução da microelectrónica mas acreditamos que a breve prazo muito mais poderá ser dito. Por agora, repetimos o que em anteriores edições mencionámos:

-Em informática, já nada nos surpreende.

Esperamos que os nossos leitores tenham ficado com uma ideia sobre a "guerra dos Chips" e, como nós, aguardem pela próxima surpresa. Quanto a nós... esperamos tudo...

ESPAÇO DE PROGRAMAÇÃO PC

Com a abertura deste novo espaço na «RS232», pretendemos dar um apoio a todos aqueles que se dedicam à programação, a nível profissional ou não.

Neste espaço irão aparecer: Truques de programação, rotinas e pequenos programas, nomeadamente jogos. Iremos utilizar algumas linguagens de programação, tais como Assembler, C, Basic.

Para já, vamos começar com um jogo em basic PC, que poderá ser uma ferramenta de aprendizagem muito útil.

Numa primeira análise poderemos não ver qualquer utilidade (a nível de aprendizagem) no programa, no entanto se analisarmos bem poderemos ver, tais como:

- ± Rotina em Assembler para Scroll horizontal
- ± Rotina para usar o JOYSTICK
- ± Pequena biblioteca de gráficos
- ± Aprender a criar gráficos
- ± Como utilizar os gráficos

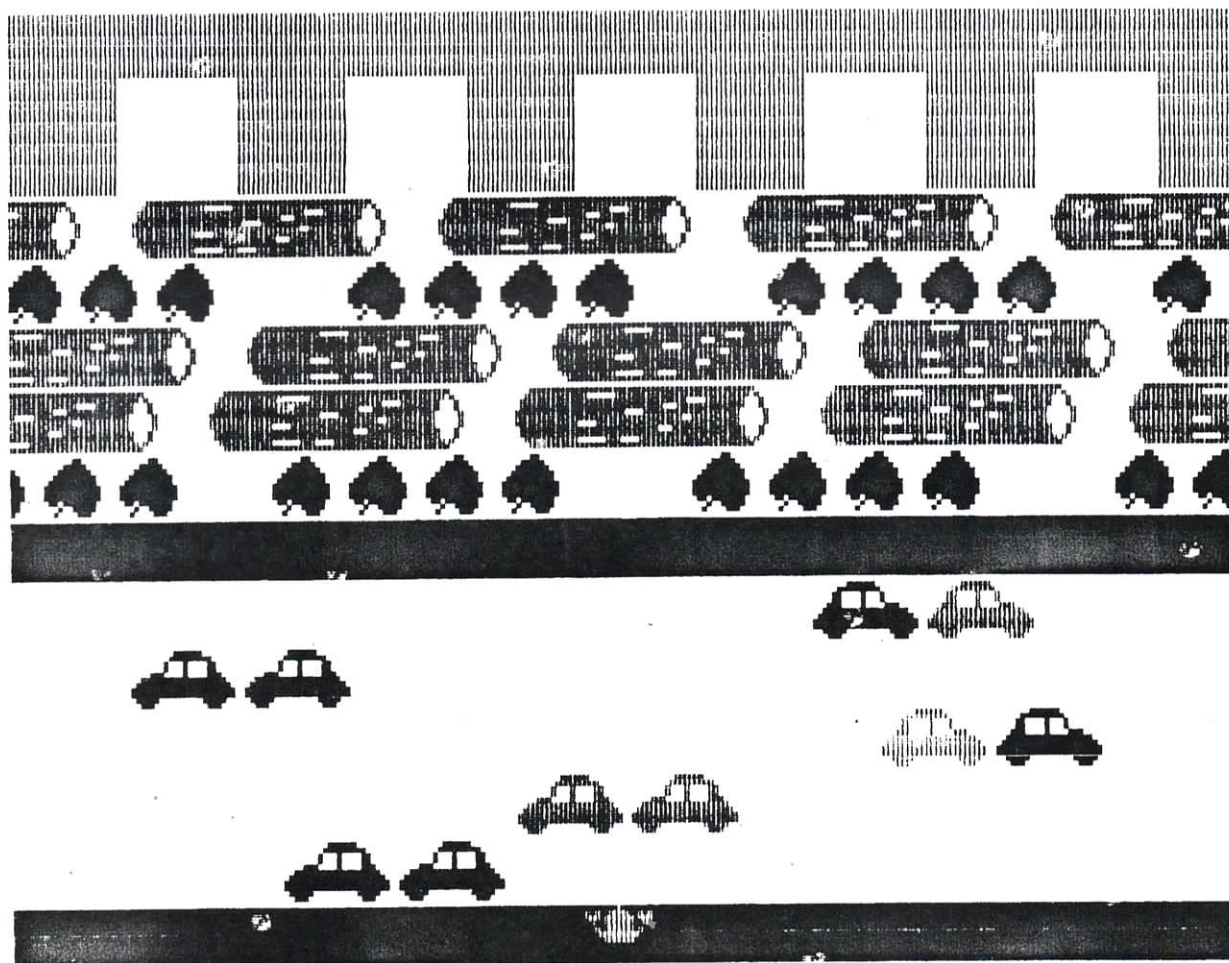
Bem, como viram, já podem utilizar a rotina do Scroll horizontal nos vossos programas, e também o vosso próximo jogo já poderá trabalhar com o JOYSTICK. Se estudarem bem este programa, os vossos conhecimentos aumentarão, estou certo disso.

Para o próximo número vamos trazer uns pequenos truques de programação.

Até lá...boa aprendizagem.

Jorge Gomes Fonseca

ECRAN DO JOGO:



Pontos: 0

Tempo: 78

```

1 `-----
2 `=      Passagem do Jogo "FROGGER" para Basic PC      =
3 `=      Copyright JOGOFO Software © 1990              =
4 `=      Autor: Jorge Gomes Fonseca                    =
5 `-----
6 `
7 `      Esta listagem é compatível com todo o tipo de Basic PC
8 `      Basic IBM, GWBASIC, MSBASIC, Turbo Basic, etc.
9 `
10 `
11 `----- limpeza e estrutura da memória
12 ` CLEAR, &H8000: DEF FNUP$(X$) = CHR$(ASC(X$) + 32 * (ASC(X$) > 95)): DIM CARRO$(1), DA(12), TS(9), TN$(9), BK%(23)
15 `----- Gráficos
20 SAPO$ = "C1RFL3BL3L0BL2R0BR11R0BR2DL2BL2L5BL2L2FBR3R5BR3GL0BL2L5BL2FR7GL5R5BFBRL0BL2L5BL2DR9DBL3
    L3BL3DL2BR11R2"
30 MACA$ = "C3 F3DFD2GDGL2H2UE2G3HBD2DOGBU2LHU2E4RE"
40 ROLO$ = "C2L45G2DGD2FDF2R46E2UEU2HUH2G2DGD2FDF2"
45 TXTR$ = "COBU3L3BD3L1BH2L2BG1BL4L3BH3L5BD3BG2R3BG3R5BR4R3BE2BR3R2"
50 CARRO$(0) = "R5FL8GRBR5R0BR4DBL4L0BL5LGR2FRL17GR19FL21DR21BDBLL4BL10L4BFBR2L2BR14R2BR2BE10"
60 CARRO$(1) = "L5GR8FLBL5L0BL4DBR4R0BR5RFL2BL5L0BL5L2GLR17FL19GR21DL21BDBRR4BR10R4BGBL2R2BL14L2BR26
    BE10"
65 `----- Carregamento em memória da rotina de Scroll horizontal -----
70 DEF SEG : CS = INT ((PEEK(778) + 256 * PEEK(779) + 514) / 16) : DEF SEG = 0: CS = CS + PEEK(&H510) + 256 * PEEK(&H511) +
    1: DEF SEG = CS: FOR X = 0 TO 231: READ N: POKE X, N: NEXT X: HS = 0
80 RANDOMIZE VAL(RIGHT$(TIMES$, 2)) + 100 * VAL(MID$(TIMES$, 4, 2))
85 `----- Passagem para o modo gráfico 320x200 4 cores CGA -----
90 SCREEN 0: SCREEN 1, 0: COLOR 0, 0: KEY OFF: CLS
92 `----- Periférico a utilizar -----
95 LOCATE 11: PRINT "Joystick ou Teclado ( prima -> J/T )": : JF = 0
100 A$ = FNUP$(INPUT$(1)): IF A$ <> "J" AND A$ <> "T" AND A$ <> "j" AND A$ <> "t" THEN 100
110 IF A$ = "j" OR A$ = "J" THEN JF = -1
120 CLS : PRINT "Inicializando...": KP$ = "HMPK": NK$ = "8624"
130 DA(1) = 8: DA(2) = -8: DA(3) = 16: DA(4) = -8: DA(5) = 16: DA(6) = 0: DA(7) = 8: DA(9) = -8: DA(10) = 16: DA(11) = -16: DA(12) =
    -8
140 OPEN "SAPO.REC" FOR RANDOM AS 1: CLOSE : OPEN "SAPO.REC" FOR INPUT AS 1
150 X = 0: WHILE NOT EOF(1): INPUT #1, TS(X): LINE INPUT #1, TN$(X): X = X + 1: WEND: CLOSE
155 `----- Desenho do ecran de acção -----
160 CLS : LINE (0, 0) -(255, 33), 1, BF: GOSUB 610: LINE (266, 0)-(319, 177), 1, B: LINE (267, 1)-(318, 176), 1, B: LINE (0, 95)-(255, 105),
    , BF: LINE (0, 167)-(255, 177), , BF
170 SC = 0: FR = 5: B = 1: LOCATE 24: PRINT SPC(4); "Pontos: 0"; SPC(8); "Tempo: "; : FOR Y = 0 TO 3: PSET (291, Y * 40 + 20): DRAW
    SAPO$: NEXT
180 FA = 0: LG = 4 + (B > 1) + (B = 4): LP = 4 - B - (B = 4)
184 FOR Y = 35 TO 83 STEP 12
185 IF Y = 35 OR Y = 59 OR Y = 71 THEN SP = INT (INT(263 / LG) / 8 + .5) * 8: ID = 57 * (LG = 2) + 50: FOR X = ID TO ID + (LG - 1) *
    SP STEP SP: PSET (X, Y): DRAW ROLO$: PAINT (X - 9, Y + 5) , 2, 2: DRAW TXTR$: NEXT: GOTO 200
190 SP = INT(INT(256 / LP) / 8 + .5) * 8: ID = -56 * (LP = 2) - 112 * (LP = 1) + 8: FOR X = ID TO ID + (LP - 1) * SP STEP SP: FOR D = 6
    TO 54 STEP 16: PSET (X + D, Y): DRAW MACA$: PAINT (X + D, Y + 3), 3, 3: NEXT D, X
200 NEXT Y: FOR Y = 8 TO 12: RD = INT(RND(1) * (200 - B * 24) + 20): PSET (RD, Y * 12 + 11): C = -(SGN(DA(Y)) < 0): FOR X = 0 TO
    B: DRAW "C" + STR$(INT(RND(1) * 3) + 1) + CARRO$(C): NEXT X, Y
210 DEF SEG = 0: POKE 1050, PEEK(1052): FOR D = 1 TO 100: NEXT: LOCATE 24, 29: PRINT "80": : FX = 120: FY = 12: TM = 80
220 YA = 23 + FY * 12: GET (FX, YA)-(FX + 13, YA + 10), BK%: PSET (FX + 6, YA): DRAW SAPO$
230 IF FY > 6 AND FY < 12 THEN FOR X = 2 TO 23: IF BK%(X) THEN 430 ELSE NEXT
240 IF FY < 6 THEN CPL = 0: FOR X = 2 TO 22 STEP 2: CPL = CPL - (BK%(X) <> 0): NEXT: CPR = 0: FOR X = 3 TO 23 STEP 2: CPR
    = CPR - (BK%(X) <> 0): NEXT: IF CPL < 4 OR CPR < 4 THEN 430
250 IF FY = 6 OR FY = 12 THEN FOR X = 1 TO 200: NEXT
260 LOCATE 24, 28: PRINT TM: : SOUND 300, .1: M$ = INKEY$: IF M$ = "" THEN M = 0 ELSE M = ASC(M$): IF M = 0 THEN M = INSTR
    (KP$, MID$(M$, 2)): IF M = ASC(MID$(NK$, M))
270 IF M = 27 THEN A$ = INPUT$(1): GOTO 260
280 IF NOT JF THEN 310

```

>>>>>>



O COMMODORE *AMIGA 500* permite
gráficos sensacionais.

*Paleta de 4096 cores - 32 cores possíveis
com uma resolução de 320 pixel.*

4 canais de som estéreo.

*Sendo multitarefa, possibilita várias
tarefas em simultâneo.*

Consulte um distribuidor Autorizado

COMMODORE AMIGA 500

 **Commodore**

Commodore Portuguesa Electrónica, S. A.
Praça Infante D. Pedro
Lote 13 - B r/c Dto.
Miraflores
1495 LISBOA
Telef. 410 11 16 - Fax, 410 83 11

```

285 `----- Rotina principal -----
290 M = 0: PO = STICK(1): P1 = STICK(1): CO = 30 < P0 < 255: C1 = 30 < P1 AND P1 < 255: IF C1 AND NOT C0 THEN M = 52 - 2* (P0
    > 127): GOTO 310
300 IF CO AND NOT C1 THEN M = 56 + 6 * (P1 > 127)
310 IF M = 50 OR M = 52 OR M = 54 OR M = 56 THEN SOUND 400, .5: SOUND 600, 1: IF M = 56 THEN SI = 20: GOSUB 600: GOTO 330
    ELSE IF M = 50 AND FY < > 12 THEN SI = -20: GOSUB 600: GOTO 330
320 FOR D = 1 TO 50: NEXT: `espera
330 NX = FX: NY = FY: IF M = 56 THEN NY = FY - 1
340 IF M = 52 AND FX - 16 >= 0 THEN NX = FX - 16
350 IF M = 54 AND FX + 16 < 240 THEN NX = FX + 16
360 IF M = 50 AND FY < > 12 THEN NY = FY + 1
370 IF NY THEN 390
380 PUT (FX, 35), BK%, PSET: GET (FX, 23)-(FX + 13, 33), BK%: PSET (FK + 6, 23): DRAW SAPOS: FOR X = 2 TO 23: IF BK%(X) <>
    0 THEN YA = 23: GOTO 430 ELSE NEXT: GOTO 580
390 IF FY > 6 THEN PUT (FX, YA), BK%, PSET: GOTO 420
400 NX = NX - DA(FY) * (M < > 50): IF NX < 0 OR NX > 239 THEN 430
410 IF M THEN PUT (FX, YA), BK%, PSET
420 DEF SEG = CS: CALL HS: FX = NX: FY = NY: TM = TM - 1: IF TM THEN IF M OR FY > 6 THEN 220 ELSE 230
430 FOR X = 700 TO 300 STEP -20: PSET (FX + 6, YA): DRAW SAPOS: PUT (FX, YA), BK%, PSET: SOUND X, 1: NEXT
440 FR = FR - 1: IF FR THEN LINE (285, FR * 40 - 20)-(298, FR * 40 - 10), 0, BF: GOTO 210
450 FOR X = 1 TO 500: NEXT: CLS : FOR X = 1 TO 4: SOUND 100, 3: SOUND 300, 3: NEXT: PRINT : PRINT : PRINT TAB(12); "GAME
    OVER": PRINT : PRINT : IF TS(0) = 0 AND SC = 0 THEN 510
460 IF SC <= TS(9) THEN 490
470 PRINT "Conseguiu bater o Record": PRINT : DEF SEG = 0: POKE 1050, PEEK(1052): LINE INPUT "Introduza o seu nome: "; TN$(9):
    TS(9) = SC: FOR X = 0 TO 8: AP = X: FOR Y = X + 1 TO 9: IF TS(Y) > TS(AP) THEN AP = Y
480 NEXT Y: HN = TS(X): HN$ = TN$(X): TS(X) = TS(AP): TN$(X) = TN$(AP): TS(AP) = HN: TN$(AP) = HN$: NEXT X: PRINT : PRINT
490 FOR X = 0 TO 9: IF TS(X) = 0 THEN 510
500 PRINT USING "##### "; TS(X); : PRINT TN$(X): NEXT
510 PRINT : PRINT : PRINT "Deseja jogar outra vez ao Sapo? (S/N) "; : CF = 0
520 A$ = INKEY$: IF A$ = " " THEN CF = CF + 1: IF CF = 20 THEN PRINT CHR$(95); CHR$(29); : GOTO 520 ELSE IF CF = 40 THEN
    CF = 0: PRINT " "; CHR$(29); GOTO 520 ELSE 520
530 A$ = FNUP$(A$): IF A$ <> "S" AND A$ <> "s" AND A$ <> "n" AND A$ <> "N" THEN 520
540 IF A$ = "S" OR A$ = "s" THEN 160
550 ON ERROR GOTO 570
555 `----- Introdução da pontuação -----
560 OPEN "sapo.rec" FOR OUTPUT AS 1: FOR X = 0 TO 9: PRINT #1, TS(X): PRINT #1, TN$(X): NEXT X
570 ON ERROR GOTO 0: CLOSE : CLS : SYSTEM
580 SI = 250: GOSUB 600: FA = FA + 1: FOR X = 250 TO 350 STEP 10: SOUND X, 1: NEXT X: IF FA < > 5 THEN 210
590 SI = 2000: GOSUB 600: PLAY "P2L8C.CL16CL8D.GL16FL8EL4C": B = B + 1 + (B = 4): GOSUB 610: GOTO 180
600 SC = SC + SI: LOCATE 24, 12: PRINT SC; : RETURN
610 FOR X = 23 TO 215 STEP 48: LINE (X, 12)-(X + 24, 33), 0, BF: NEXT: LINE (0, 35)-(255, 94), 0, BF: LINE (0, 106)-(255, 166), 0, BF:
    RETURN
615 `----- Rotina em Assembler do Scroll horizontal -----
620 DATA 235,18,144,0,0,1,255,2,255,2,0,1,255,2,254,255,0,0,0,30,6,184,85,184,142,216,142,192,46,199,6,3,0,10,0,46,139,54,3,0,46,138,
    132,5,0,60,0,117,3,233,159,0,46,162,17,0
630 DATA 46,162,18,0,139,198,187,224,1,247,227,139,216,46,198,6,16,0,2,46,246,6,17,0,128,117,66,253,131,195,62,139,251,46,198,6,19,0,
    6,139,247,131,238,2,139,5,185,31,0,243,165,137,68,2,129,199,142
640 DATA 0,46,254,14,19,0,117,230,46,254,14,17,0,117,215,46,254,14,16,0,116,76,46,160,18,0,46,162,17,0,129,195,0,32,235,194,252,139,
    251,46,198,6,19,0,6,139,247,131,198,2,139,5,185,31,0,243,165
650 DATA 137,68,254,131,199,18,46,254,14,19,0,117,231,46,254,6,17,0,117,216,46,254,14,16,0,116,14,46,160,18,0,46,162,17,0,129,195,0,32,
    235,195,46,255,14,3,0,46,129,62,3,0,255,255,116,3,233,63,
660 DATA 255,7,31,203

```

JORGE GOMES FONSECA



PAPELARIA FERNANDES **INDÚSTRIA E COMÉRCIO, S.A.**

Irá abrir no próximo mês de Dezembro, no Lg. do Rato 14, a sua loja de **Informática** onde poderá encontrar:

PC's XT
PC's AT a 12MHz ; 20MHz
PC's 386 a 33MHz
PORTÁTEIS
TRANSPORTÁVEIS
IMPRESSORAS
MONITORES
CONSUMÍVEIS
SOFTWARE EDUCATIVO
SOFTWARE CAD

LISBOA -SEDE	Largo do Rato, 13 * Tel.: 682131 - 693048 * 1296 LISBOA CODEX
SEC. TÉCNICA	Rua Saraiva de Carvalho, 145 * Tel.: 658141 - 691046
FILIAIS-LISBOA	Rua do Ouro, 145 * Tel.: 320091 Rua do Crucifixo, 62 * Tel.:364152 - 368541
PORTO	Praça Marquês de Pombal, 215 * Tel.: 489068 - 484551 R. dos Bragas, 304 * Tel.:382936 - 380421 - 382827 - 4000 PORTO
COIMBRA	Rua Castro Matoso, 12 C-D * Tel.: 27720 Av. Dr. Urbano Duarte, 22 * Tel.: 721789 - 3000 COIMBRA
SETÚBAL	Rua Antão Girão, 81 - 83 * Tel.: 33474 Tv. do Corpo Santo, 8 * Tel.: 24510 - 2900 SETÚBAL
SANTARÉM	Rua 16 de Abril, 6 (Alto do Bexiga) * Tel.: 20361 - 2000 SANTARÉM

INTRODUÇÃO AO CÓDIGO MÁQUINA

PARTE III - Como funciona o Z80

4.3. - As mnemónicas do Z80 (Continuação)

GRUPO 7 - As instruções de subtracção

Também neste grupo vão aparecer três tipos diferentes de mnemónicas, que iremos tratar separadamente. Estas, são similares às do grupo 6, mas efectuando uma operação de sinal contrário. Assim, podemos estabelecer a seguinte ordem de comparação:

No primeiro subgrupo:

Grupo 6 - Somar s/ Carry - ADD
Grupo 7 - Subtrair s/ Carry - SUB

No segundo sub grupo:

Grupo 6 - Somar c/ Carry - ADC
Grupo 7 - Subtrair c/ Carry - SBC

No terceiro sub grupo:

Grupo 6 - Incrementar (+1) - INC
Grupo 7 - Decrementar (-1) - DEC

Na adição quando o registo A ultrapassa o seu máximo de acumulação a contagem volta a zero, sendo o Carry flag impulsionado para o estado (1), acção que representa adicionar o número 256 ao novo valor encontrado no registo.

Na subtracção, quando o registo A ultrapassa, no sentido decrescente, o seu mínimo (todos os bits a zero), a contagem salta para o seu máximo de acumulação (todos os bits no estado 1) tal e qual um conta voltas mecânico. Nesse momento o Carry flag é impulsionado para o estado (ON), acção que representa subtrair o número 256 ao novo valor encontrado no registo.

Quem não tiver presente este conceito, terá dificuldade em compreender o que for explicado mais adiante.

Exemplo da passagem pelo valor máximo numa adição.

(antes) +1 Carry 0 byte 11111111
(depois) Carry 1 byte 00000000
(256 * 1) + 0 = 256

Exemplo da passagem pelo valor mínimo numa subtracção:

(antes) -1 Carry 0 byte 00000000
(depois) Carry 1 byte 11111111
255 - (256 * 1) = -1

Veja com atenção os ensaios seguintes.

1º Subgrupo - instruções SUB

Este conjunto de instruções destina-se apenas a operações com registos SIMPLES (todos os registos de 8 bytes) e tendo como intervenientes principal o registo A.

Assim, a mnemónica SUB foi simplificada referenciando apenas o segundo interveniente.

Antes de efectuar a operação indicada por uma destas instruções, o Z80 passa o Flag N para 1 (indicando a próxima operação de subtracção), deixando os restantes flags reflectir o estado do acumulador, após a operação executada.

Mnemónicas	Códigos	Tempos
SUB N	214, N	São iguais aos das instruções ADD.
SUB A	151	"
SUB H	148	"
SUB L	149	"
SUB B	144	"
SUB C	145	"
SUB D	146	"
SUB E	147	"
SUB (HL)	150	"
SUB (IX + d)	221, 150 + d	"
SUB (IY + d)	253, 150 + d	"

Ensaio 1 :

Mnemónicas	Códigos	Comentários
LD HL, 27100	33, 220, 105	aponta o endereço,
LD A, 100	62, 100	coloca valor de A
LD (HL), A	119	no endereço apontado
LD A, 255	62, 255	novo valor para A
LD E, 50	30, 50	valor para E
SUB E	147	(A - E)
SUB 100	214, 100	(A - N)
SUB (HL)	150	(A - (HL))
LD B, 0	6, 0	resultado é
LD C, A	79	transferido para
RET	201	o Basic

Resposta: Conteúdo do registo A = 5

Exemplifica-se neste ensaio, utilizando instruções SUB deste subgrupo, 3 subtracções sucessivas ao conteúdo do registo A. O resultado está correcto porque o valor de A nunca deixou de ser positivo. Vejamos a seguir como o uso incorrecto da instrução SUB vai falsear um resultado.

Ensaio 2 :

Mnemónicas	Códigos	Comentários
LD A, 20	62, 20	A subtracção é incorrecta porque E > que A e o Carry não entrou na operação.
LD E, 30	30, 30	
SUB E	147	
LD B, 0	6, 0	
LD C, A	79	
RET	201	

Resultado esperado : - 10

Resultado obtido : 246

Analisando o estado dos Flags antes e depois da operação SUB, verifica-se que o Z80 foi AVISADO do resultado não ser correcto e que tomou providências para poder reconhecer o resultado real, porém não o fez porque a sequência de instruções que o poderia solicitar, não foi introduzida pelo programador.

Estado dos flags antes da operação SUB (posição 1) e após a operação (posição 2):

Flags S Z X H X P/O N C

Pos. 1 0 0 1 0 1 0 1 0

Pos. 2 1 1 0 0 1 1 0 1

Que nos dizem os flags?

Nesté caso apenas nos interessa a análise dos flags (S) e (C).

1) Flag C - O Carry Flag indica que foi ultrapassado o mínimo acumulável no registo interveniente.

2) Flag S - O Sinal-Flag indica que o número contido no registo interveniente é negativo e que se encontra sobre a forma de complemento por dois.

Ora nós já falámos sobre o 2º complemento aritmético e as suas formas de representação.

Sabemos que esses números estão compreendidos entre (-127 a +128) e que o 8º bit indicará o sinal respectivo.

Então:

O número 246 (na base 10) = ao binário (1) 111 0110

(1) - Bit de sinal indicando um resultado negativo.

Se retirarmos uma unidade ao número Binário,

$$1110110 - 1 = 1110101$$

e negarmos o resultado;

$$(1) \overline{1110101} = (-) 0001010$$

verifica-se que o resultado real é (-) 10; ou ainda, por um processo muito mais rápido:

$$246 - (1 * 2 ^ 8) \\ = 246 - (1 * 256) = - 10$$

A forma de forçar a máquina a devolver ao Basic um resultado NEGATIVO, envolve instruções de que ainda não falámos, bem como a manipulação das rotinas de cálculo com sistema de numeração em ponto flutuante situadas na ROM, que estudaremos em próximo capítulo.

Utilizando a instrução SBC, poderá o leitor para já contornar o problema, tendo sempre em conta que numa subtracção com este tipo de instrução o resultado será devolvido em 2º complemento aritmético, o que implica uma negação do número binário e considerar o tipo de sinal do oitavo bit.

2º Subgrupo - As instruções SBC

Também aqui o Flag N é comutado para 1 e os restantes flags reflectirão o estado do registo interveniente após a operação. Os tempos de execução são iguais aos das instruções ADC (ver grupo 6).

Mnemónicas	Códigos
SBC A, + N	222, + N
SBC A, A	159
SBC A, H	156
SBC A, L	157
SBC A, B	152
SBC A, C	153
SBC A, D	154
SBC A, E	155
SBC A, (HL)	158
SBC A, (IX + d)	221, 158 + d
SBC A, (IY + d)	253, 158 + d
SBC HL, HL	237, 98
SBC HL, BC	237, 66
SBC HL, DE	237, 82
SBC HL, SP	237, 114

Já aqui foi afirmado, mas repete-se dada a sua importância, que APENAS o registo A ou o par HL foram concebidos no Z80 para efectuarem operações aritméticas ou lógicas.

Na elaboração de um programa em Código Máquina, deve-se assegurar a cada momento uma certa disponibilidade destes registos, pois é necessário ter em conta que um computador numérico executa todo o tipo de cálculo tendo por base a sua manipulação binária.

Ensaio 3: Subtracção em que o número maior se encontra no registo DE e o menor, em HL.

```
LD HL, 2700 33, 10, 140 Colocação dos registos
LD DE, 5200 17, 20, 80 nas condições propostas

AND A 167 Carry reset "0"

SBC HL, DE 237, 82 subtracção com Carry
LD A, L 125 Lowbyte é passado para o
LD (32000), A 50, 125, 0 endereço 32000
LD A, H 124 Hiyte para o endereço
LD (32001), A 50, 125, 1 32001
RET 201
```

(Programa em Basic) :

```
CLEAR 29999
REM aqui instalar o carregador como em exemplos anteriores
RANDOMIZE USR 30000
LET v = PEEK (32000) + PEEK (32001) * 256
REM O número está em 2º complemento, negar o resultado.
```

REM Simplificando o calculo:

```
LET R = v - 65535
```

e irá obter o resultado final de -2500.

Vamos ver então como se processou esta operação que por vezes parece complicada. Quando o registo de 16 bits (HL) passou, na subtracção, pelo valor zero, saltou automaticamente para o seu máximo valor (tudo a uns - ou seja 65535) e continuou a ser decrementado até chegar ao valor de 63035, tendo o Carry Flag assinalado essa passagem.

Ensaio 4 - Subtrair B do conteúdo de HL

```
LD HL, 29000 33, 72, 113 Colocação dos registos nas
LD (HL), 200 54, 200 condições propostas
LD B, 100 6, 100
LD A, (HL) 126
SUB B 114

LD (HL) 119
RET 201
```

(Programa em Basic) :

```
LEAR 27999
```

REM Com o carregador habitual introduza o C.M.

```
RANDOMIZE USR 28000
```

```
PRINT PEEK 29000
```

(Resultado: 100)

Ensaio 5 - Outra subtracção cujo resultado é um valor negativo

```
AND A 167
```

```
LD A, 20 62, 20
```

```
LD E, 30 30, 30
```

```
SUB E 147
```

número negativo E > A

```
LD (29000), A 50, 72, 113
```

em complemento por 2

```
LD A, 0
```

```
SBC A, A 159
```

(A - CARRY) = 255

```
LD (29001), A 50, 73, 113
```

(Programa em Basic) :

```
CLEAR 27999
```

REM Com o carregador habitual instale o C.M.

```
RANDOMIZE USR 28000
```

```
LET N = PEEK 29000
```

```
IF PEEK 29001 = 255 THEN GOTO (X)
```

```
PRINT N
```

```
STOP
```

(X) LET K = 1 * 2 ^ 8

```
PRINT N - K
```

```
STOP
```

(Resultado: - 10)

Boas Subtracções... (CONTINUA)

FERNANDO PRECES

J. P. R. Informática

DISKETTES 3.5 DD

(PREÇOS ESPECIAIS PARA REVENDA)

SONY ; MITSUBISHI ; TDK ; KAO "COLOR" ; 3M ; VERBANTIM ; MAXELL
POLAROID ; BASF ;

(Outras diskettes sem marca, às côres)

Revenda de Software

R. GENERAL ALVES ROÇADAS 38 - 40 ; Loja 10 Centro C. ACSANTOS Telefone / Fax: 01 - 987 72 91
2675 ODIVELAS

CLUBE DE LEITORES - «RS232»

Caros amigos:

Como dissemos na edição anterior, esta «secção» será uma maneira simpática de retribuirmos o apoio que nos tem dado, com a sua preferência.

Porque ainda estão a decorrer acordos com várias empresas, a actual lista de produtos não será muito extensa.

No entanto, demonstra a nossa intenção de "começar"...

Queremos, desde já, deixar claro que para além dos produtos mencionados poderemos tentar arranjar-lhe outros, desde que tenhamos conhecimento da marca e da empresa que os comercializa.

Também queremos alertá-lo para o facto dos preços mencionados poderem vir a sofrer alteração, tanto para mais como para menos (?), entre o espaço decorrente entre o seu pedido e a entrega dos mesmos. Nestas situações, procederemos como será lógico...(cobraremos o excesso e devolveremos a quantia não necessária).

Os prazos de entrega dependem da rapidez com que nos sejam entregues os produtos solicitados pelos leitores.

Caso exista algum contratempo, informaremos de imediato.

Os pedidos podem ser efectuados através de carta e acompanhados de cheque ou vale postal no valor correspondente ao produto ou produtos a fornecer.

No caso de fornecimento de equipamentos, os portes dependerão do peso dos mesmos, sendo esse valor cobrado na altura da entrega.

Software Original ...tão fácil
...tão útil
...tão barato

DICIONÁRIO TRADUTOR I®

© JOGOF Software

...desde o Estudante até ao mais sofisticado Tradutor de línguas.

Uma ferramenta indispensável, para todos.

Nível 1 : 119.000\$00 + IVA

Tradutor I + Processador de texto com tradução + DiciCria (utilitários para criar novos dicionários e construir : plural e feminino automáticos, compactação, correcção de erros, visualização, etc.)+ Dicionário de Português-Inglês + Manuais

Nível 2 : 52.000\$00 + IVA

Tradutor I + Processador de Textos com tradução + Dicionário de Português-Inglês + Manuais

Nível 3 : 25.000\$00 + IVA

Tradutor I + Dicionário de Português-Inglês

Versão Demo: 2.500\$00+IVA

Tradutor Demo + Dicionários Demo de Português-Alemão + Francês-Português + Inglês-Português + Manuais.

Versão Demo VideoShow: 3.200\$00 + IVA

Demonstrativo com animação e imagens digitalizadas para placas gráficas VGA ou MCGA.

Pedidos a : «RS232-Informática»

Distribuidor Exclusivo para:

Lisboa / arredores e Zona Sul do País.

CLUBE DOS LEITORES «RS232»

DISQUETES

Cx de 10 Unidades

3 1/2 DD - 135 TPI MEMORY	3.625\$
3 1/2 DD - 135 TPI SELECT	3.000\$
3 1/2 DD - 135 TPI PHILIPS	2.750\$
3 1/2 HD - 135 TPI MEMORY	6.360\$
3 1/2 HD - 135 TPI SELECT	6.500\$
3 1/2 HD - 135 TPI PHILIPS	6.130\$
5 1/4 DD - 48 TPI MEMORY	980\$
5 1/4 DD - 48 TPI PHILIPS	1.450\$
5 1/4 HD - 96 TPI MEMORY	2.650\$
5 1/4 HD - 96 TPI SELECT	3.010\$
5 1/4 HD - 96 TPI PHILIPS	2.560\$
3 1/2 DD - 135 TPI RPS (cx 20)	7.415\$
3 1/2 HD - 135 TPI RPS	1.272\$
5 1/4 DD - 48 TPI MOORE	2.000\$
5 1/4 HD - 96 TPI RPS	3.890\$
3 1/2 DD - 135 TPI BOEDER	4.360\$
3 1/2 HD - 135 TPI BOEDER	9.730\$
5 1/4 DD - 48 TPI BOEDER	1.600\$
5 1/4 HD - 96 TPI BOEDER	4.460\$

FITAS / IMPRESSORA

PHILIPS PP 402	4.175\$
PHILIPS NMS1440/1441	2.000\$
PHILIPS NMS 1432	1.290\$
PHILIPS NMS 1437	1.875\$
PHILIPS NMS 1460/61	2.200\$
PHILIPS NMS 1466/7	2.200\$
CITIZEN SWIFT 24	1.545\$
SEIKOSSHA GP 500	1.665\$
AMSTRAD DMP 3000/3160	1.075\$
AMSTRAD DMP 4000	1.360\$
CIT120D/180E/LSP100	905\$
SEIKOSHA 1000/1200 - PCW	1.010\$
ADMATE/UC. CPA/880/130	840\$
MSP A3	1.150\$
HQP 45	985\$
de COR SWIFT 24	4.455\$
de COR PRODOT A4	5.570\$
de COR PRODOT 9X	6.125\$

UPGRADE AMIGA 500

512K MEMORY c/ relógio	25.000\$
512K MEMORY s/ relógio	20.000\$

PAPEL CONTÍNUO

500 FLS - 12" / 9 1/2"	940\$
2000 FLS - 12" / 9 1/2"	3.780\$
2000 FLS - 12" / 15"	5.650\$
4000 FLS - 6" / 9 1/2"	6.005\$
FAX - ROLO 30	1.050\$

Software SPECTRUM

JOGOS DIDÁTICOS: (400\$ cada)

ASTOR MÚSICA
 SUB-ROTINAS DE MATEMÁTICA
 GEOGRAFIA DE PORTUGAL
 TABELA PERIÓDICA
 MATEMÁTICA INFANTIL
 GRÁFICOS DE FUNÇÕES
 HISTÓRIA UNIVERSAL
 INVENTOS E INVENTORES
 CÁLCULOS DE ÁREAS E VOLUMES
 O ESQUELETO HUMANO
 O APARELHO DIGESTIVO
 UNIVERSO
 GRAND. REGIÕES BIO-CLIMÁTICAS
 O SISTEMA SOLAR
 DIV. E CLASSIF. DE SERES VIVOS
 AS PLANTAS
 GENÉTICA
 ORGÃOS REPRODUTORES
 GEOLOGIA

JOGOS DIVERSOS (200\$ cada)

ASTOR MIND
 BATALHA NAVAL
 BLOCO MÁGICO
 BIOCAL (Biorritmo/Calendário)
 BONG
 DOCE LAR
 FÉRIAS EM PORTUGAL
 FOGUETÃO
 GUERRA KRIPTO
 ASTOR E OS BIDONS
 ASTOR NA LUA

JOGOS DE CASINO (200\$ cada)

ASTOR ROLETA
 BINGO
 MÁQUINA DE POKER
 POKER ABERTO
 ASTOR GAMÃO
 ASTOR SLOT MACHINE

Utilitários SPECTRUM

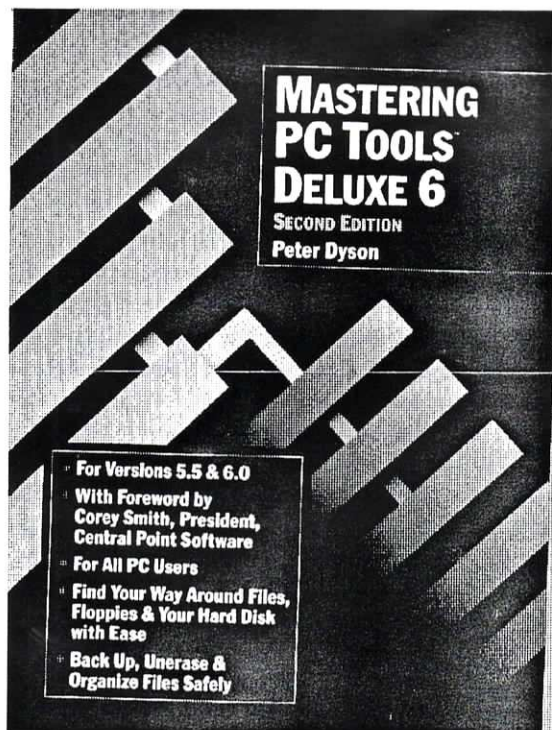
Análise Estrutural	400\$
Investimentos	400\$
Basic - Facts -Biology	450\$
Phisics	450\$
Chemistry	450\$
Beta Basic 3.0	1.000\$
Contas Correntes	400\$
Decision Maker	600\$
Dev Pac 4	750\$
Dlan	450\$
Estrutura Atômica	450\$
Biblioteca	400\$
Forth	500\$
Full Screen	500\$
Funções	400\$
Graphic Adv.Creator	350\$
Geography	400\$
Geometria analítica	400\$
Geometria Descritiva	500\$
Gestão Comercial	1.000\$
Gestão de Stocks	1.000\$
HI-T	400\$
Indice Bibliográfico	400\$
Keydefine	400\$
Electronics	500\$
Genius (Laser)	750\$
Compilador C	1.000\$
Masterfile	500\$
Master Toolkit	500\$
Matrizes-Matemática	400\$
Mega Basic	500\$
Prolog	500\$
Music Typwrite	500\$
PaintBox	500\$
Pert	400\$
Screen Machine	500\$
Snail Logo	500\$
Spectrum Monitor	400\$
Star Watcher	500\$
Stocks	400\$
Stocks Manager (80 cl)	500\$
Supercode 3.5	500\$
Tasmerge	500\$
Tasprint	500\$
The last word	500\$
Word processor	500\$
Vat Manager (80 cl)	500\$
VU - 3D	400\$
VU - Calc	400\$
VU - File	400\$

Ao seu pedido, deve fazer acompanhar cheque ou vale postal acrescido dos portes.

Como já divulgámos em anterior edição, o número de títulos excede os 900 pelo que se torna impossível inseri-los. Por este facto, pedimos aos interessados que nos solicitem a referida lista.

NOVOS TÍTULOS:

MASTERING PC TOOLS DELUXE 6



PC Tools Deluxe 6, é um dos programas mais utilizados pelos técnicos e utilizadores de micro informática. É para esses profissionais que a SYBEX acaba de editar este completo guia de referência destacando esta última versão Deluxe 6, que dadas as similaridades com o já popular Pc Tools Deluxe 5.5, expondo sempre as respectivas diferenças, torna-se num manual de mais vastas aplicações.

O autor começa por transmitir ao leitor uma noção breve do computador e da estrutura do DOS, tornando este guia muito prático, contendo abordagens muito pormenorizadas sobre o manuseamento de ficheiros e directorias, dando especial relevo não só às questões de segurança e recuperação de dados, como também ao campo das telecomunicações que tem como novidade o funcionamento com placas FAX.

Será ainda este manual muito útil, porque ensina a configurar e adaptar a si próprio todo o funcionamento deste imenso conjunto de utilitários, que transformam o seu computador num mundo de aplicações proporcionadas por este novo Pc Tools Deluxe 6.

PREÇO - 6.840\$00 / Assinantes RS232 = 10% desc.

STORYBOARD PLUS

Finalmente em Português o primeiro livro que se dedica da forma mais completa à utilização do Storyboard Plus, software de IBM de grande aceitação no mercado internacional.

Este manual tem como objectivo dar ao leitor os meios de alcançar aos melhores resultados utilizando todo o potencial deste software.

Sessões de apresentação, demonstrações, relatos de acontecimentos, etc., e todo o tipo de aplicações em que se exija uma ilustração com figuras e legendas, ou numa simples mostragem de imagens em sequência, constituem entre outras as aplicações que este software pode proporcionar.

Esta obra ajuda ainda a tornar mais próxima e directa a comunicação entre o Homem e a máquina, ensinando a utilizar os periféricos (teclado, mouse, joystick, mesa digitalizadora) admitidos pelo programa.

PREÇO : 6.035\$00 / Assinantes RS232 = 10% desc.



GESTÃO DA FACTURAÇÃO DE UM EMPRESA

* Implementação do programa de gestão de facturação.

* Algumas características da programação estruturada em C.

Apresentei no artigo anterior a primeira parte da listagem respeitante à implementação do nosso programa. Nela foram implementados os primeiros passos a serem tomados sempre que se elabora um programa em C:

- * inclusão das bibliotecas necessárias ao funcionamento do programa;
- * definição de macros e de tipos de dados;
- * definição de dados estruturados;
- * declaração de variáveis globais;
- * inclusão de módulos específicos ao programa.

Expliquei ainda que um programa deve ser dividido em blocos de funções que desempenham determinadas funções específicas. No presente caso a divisão do programa foi realizada de acordo com as seguintes tarefas:

- * manuseamento do ecrã;
- * saída de dados (output);
- * entrada de dados (input);
- * gestão das estruturas de facturas e pagamentos;
- * gestão das estruturas dos clientes;
- * funções auxiliares diversas.

Os módulos que implementam estes blocos têm a eles associadas duas files: uma que contém a declaração das funções desse módulo e cuja extensão é geralmente do tipo ".h", e outra que contém o código em C que implementa essas funções e cuja extensão é do tipo ".c".

A listagem que foi apresentada encontra-se contida no módulo principal que contém essencialmente funções de controle da execução do programa. Vamos neste artigo examinar a implementação de algumas dessas funções, e cujos algoritmos foram apresentados na revista número 19 do mês de Setembro.

A listagem correspondente a esta implementação deveria ser apresentada de seguida mas, por motivos de ordem técnica e de espaço, fá-lo-emos no próximo número.

Compreendendo que "interrompemos" (não intencionalmente) o conteúdo deste tema, apresentamos as nossas desculpas ao leitor e ao autor do mesmo. Passaremos ao texto que a essa listagem se segue.

Texto após listagem:

A primeira função chama-se **main** e é aquela por onde a execução do programa começa. Neste caso a função **main** foi declarada de forma a não receber parâmetros, uma vez que não existem quaisquer argumentos entre os parâmetros colocados a seguir ao nome. O carácter '{' indica o início da função, e o seu oposto '}' indica o fim da função.

Entre estes dois caracteres podemos encontrar a declaração das variáveis locais à função, a invocação de outras funções, e instruções básicas e de controlo. No que respeita a variáveis locais, a função **main** tem apenas uma, cujo "nome" é identificado pela letra 'c'; o tipo desta variável é **char** o que quer dizer que se trata de um, e apenas um, caractere.

A primeira acção a ser tomada consiste na invocação da função **begin_system ()** que inicializa o sistema (criando ficheiros, inicializando variáveis globais, etc.). Seguidamente é invocada a função **clear ()** à qual são passados quatro parâmetros de entrada. Esta função tem por tarefa apagar uma determinada zona do ecrã consoante o valor dos parâmetros recebidos.

Os dois primeiros definem as coordenadas do canto superior esquerdo dessa zona e os dois últimos definem o canto inferior direito da mesma.

No presente caso, os quatro parâmetros (0, 0, 80, 24) definem todo o ecrã; assim a invocação **clear (0, 0, 80, 24)** resulta em apagar todo o ecrã - algo de semelhante à instrução **CLS** do **BASIC**.

A terceira instrução da função **main ()** tem por objectivo a impressão no ecrã da "máscara" principal na qual são apresentadas todas as opções disponíveis ao utilizador. Em seguida é iniciado um ciclo do tipo **do-while (fazer-enquanto)**, no interior do qual são executadas duas instruções. A primeira destas instruções consiste numa **atribuição** na qual é invocada uma função. Ou seja, é atribuído à variável "c" o valor de retorno da função **lê_opção()**. Daqui podemos concluir que esta função não tem parâmetros de entrada mas tem um parâmetro de saída. O significado do valor desse parâmetro é a opção que o utilizador pretende que seja executada. De facto, a função **lê_opção()**, lê da consola (e valida) a tecla que foi premida, devolvendo o código **ASCII** dessa tecla.

A segunda instrução do ciclo **do-while** consiste na invocação da

função `executa_opção()` à qual é passado um parâmetro de entrada: o valor da variável "c". Esta função tem por tarefa controlar a execução da opção que lhe é passada como parâmetro. Finalmente, a função `main()` termina com uma nova invocação da função `clear()` que apaga todo o ecran.

Poderemos verificar (no próximo número...) que na listagem, à direita do código em C, se encontra descrito o algoritmo sob a forma de comentários. Embora não seja frequente a descrição algorítmica das funções junto com a listagem (principalmente quando estas são extensas), neste caso pretendeu-se que o leitor tivesse mais facilidade na leitura desta.

Vamos agora examinar a implementação da função `begin_system()`, que por sinal é invocada na função `main()`. Antes do código propriamente dito, podemos encontrar uma breve especificação da função. Provavelmente o leitor não se encontra familiarizado com a notação utilizada, contudo é fácil compreender o seu significado e a utilidade de comentários que especificam as funções é indiscutível.

De resto, o programador tem a liberdade de estruturar os comentários dos seus programas da forma que achar conveniente, desde que o faça de uma forma clara e coerente.

O algoritmo que descreve a função `begin_system()` foi apresentado na revista 19.

A relação que existe entre este e o código em C que o implementa é óbvia. Não foram utilizadas regras formais para a passagem do algoritmo à sua implementação, sendo uma questão de conhecimento da linguagem utilizada, a escolha das instruções apropriadas à implementação. Se dermos agora (?) uma olhadela na listagem, verificamos que não existem variáveis locais nesta função, uma vez que nenhuma se encontra declarada. Podemos no entanto verificar que, logo no início da função, se encontram declaradas três funções:

`inic_ft_list()`, `inic_pg_list()` e `inic_free_list()`.

As duas primeiras são do tipo `*L_FACT_PAG` (o asterístico significa que se trata de um apontador), que foi definido na listagem anterior, como sendo a estrutura dos registos das listas ligadas, que indexam os ficheiros de facturas e de pagamentos.

Assim, estas duas funções devolvem um apontador para um registo numa lista de indexação. A terceira função é do tipo `void` o que significa que não devolve qualquer valor de retorno - ou seja, não tem parâmetros de saída.

Poderá o leitor interrogar-se neste momento acerca da declaração de funções dentro de outras funções. A questão é mais simples do que pode parecer à primeira vista. Para que uma função possa invocar outra função, tem que "conhecer" essa função, ou seja, saber que tipo de dados retorna. Uma função é conhecida por outra função caso tenha sido declarada "fora" e antes de qualquer função, ou caso essa sua declaração tenha sido realizada "dentro" da função que a pretende invocar.

Sempre que a declaração de uma função tenha sido realizada "fora" e antes de qualquer função, esta passa a ser "conhecida" por todas as funções do programa - algo semelhante ao que se passa com as variáveis globais. No caso da declaração ter ocorrido no "interior" de outras funções, apenas essas conhecem a função declarada.

A decisão de declarar as funções dentro ou fora das funções, depende de vários factores, uma vez que ambos os casos arrastam consigo algumas consequências.

No presente caso, a função `begin_system()` invoca as funções declaradas no seu interior, e por isso deve "conhecê-las".

A tarefa que estas funções realizam consiste na inicialização das estruturas que indexam os ficheiros existentes.

As três primeiras instruções realizam a abertura dos três ficheiros no modo de leitura (identificado pela string "r+b").

Caso algum destes ficheiros não exista, é devolvido o valor `NULL` pela função `fopen()`. Assim, caso o ficheiro de clientes "CLIENTES.FCH" não exista, a variável global `fp_cl` fica com o valor `NULL`.

De seguida é verificada a existência do ficheiro de clientes.

Caso não exista, o ficheiro é criado através da abertura em modo de escrita (identificado pela string "w+b") e é inicializada a zero a variável global `last_cliente` (que contém o código do próximo cliente a ser inserido). Se o ficheiro já existir, o valor desta variável é lido da primeira posição do mesmo ficheiro.



CARDITA - ANÁLISE INFORMÁTICA Lda

Informatização de Empresas

Software de Gestão

Package específicos para:

- Escolas, Seguros; Clubes de Video;
- Gestão de Associações.

Software Específico.

CURSOS DE FORMAÇÃO

Direct Mail

Recolha de Dados

Serviço Bureau

ÚNICA EMPRESA COM D.R.H.C

Adm. : R. Diogo Azambuja, 2 - 1400 Lisboa Tel.: (01) - 61 08 04

Escritório: Av. da República, 6-B c/v Esq. Algés - 1495 Lisboa Tel.: (01) - 211 42 62

A instrução seguinte invoca a função `inic_hashtab ()`, que inicializa a tabela de hashing de indexação ao ficheiro de clientes.

As instruções seguintes repetem estas acções, mas desta vez para os ficheiros de facturas e de pagamentos. A última instrução invoca a função `fcloseall ()` que fecha **todos** os ficheiros que se encontrarem abertos. Esta função pertence às livrarias do TURBOC, e pode ser invocada se fôr feita a inclusão da "file" `stdio.h` (standard input / output).

A função que se segue na listagem, é a `limpeza()`, e não tem quaisquer parâmetros de entrada. O **tipo** desta função é void, não havendo portanto valor de retorno. Existe novamente a declaração de uma função. Trata-se da função `begin_system()`, que acabamos de examinar. A sua declaração existe pelo facto desta função ser invocada, e como tal deve ser "conhecida" pela presente função. A tarefa realizada é muito simples e consiste em apagar os ficheiros do sistema e de seguida reinicializá-lo. Sempre que esta função é invocada, todos os dados que se encontravam presentes no sistema são destruídos, "arrancando" o programa com os ficheiros vazios. Das quatro instruções existentes, as três primeiras consistem na invocação da função **unlink** que pertence às livrarias do TURBOC, e cujo efeito é o de apagar o ficheiro definido pela string que lhe é passada como parâmetro. A última instrução é, como já vimos, uma invocação à função `begin_system` sendo então o sistema reinicializado.

O algoritmo que descreve a função `limpeza()` encontra-se descrito no artigo da revista nº19.

Trata-se do algoritmo número dois, e a sua semelhança com a respectiva implementação é óbvia.

A função `main_insert_cliente()` implementa, por sua vez, o algoritmo 3 descrito no artigo da revista nº19. Esta função tem como tarefa realizar o controle da inserção dos clientes no sistema. Trata-se pois de uma função de **controle** que interactua com o utilizador e invoca outras funções que realizam tarefas mais específicas, e portanto de mais baixo nível. Uma vez mais, todas as funções que são invocadas no interior desta função, encontram-se declaradas logo no seu início. É neste ponto um bom exercício para o leitor fazer uma comparação entre o algoritmo e a sua implementação em C.

O problema da declaração de funções no interior de outras, merece ainda uma última chamada de atenção.

Vimos atrás que quando uma determinada função pretende invocar outra, deve conhecer essa "outra". As duas possibilidades que asseguram que uma função `X()` conheça uma outra função `Y()` são as seguintes:

ou a função `Y()` tinha sido declarada **fora e antes** da função `X()`, ou a função `Y()` é declarada dentro da função `X()`. Ora, logo no início do programa, antes do código C das funções, é feito o "include" das files ".h" que contêm a declaração das funções dos módulos a que dizem respeito.

A função `begin_system()`, por exemplo, encontra-se declarada na file "gfmmain.h" e portanto todas as funções do nosso programa a conhecem.

De resto, todas as funções existentes (excepto a função `main()`) encontram-se declaradas numa destas files. Assim sendo, não há de facto a necessidade de funções serem declaradas no interior de outras.

Contudo, pelas mais diversas razões, pode acontecer da declaração de uma função ser realizada **após** a função que a pretende invocar, ou pode até não ser declarada.

Por vezes, ligeiras alterações na estrutura de um programa podem originar situações deste tipo.

Portanto, é até certo ponto uma segurança declarar funções dentro de outras. De resto, tal atitude "obriga" o programador a ter a listagem dum programa mais coesa e estruturada.

Por exemplo, se quisermos saber quais as funções que invocam uma determinada função em particular, basta procurar as funções que no seu início a declaram.

No entanto, se num programa tudo estiver claro e sob controle do programador, não há geralmente a necessidade de declararmos funções no interior de outras.

Esta questão tem também a ver com a filosofia de programação que se segue, o que está de algum modo relacionado com o estilo do programador. Se uma linguagem C permite a declaração de funções no interior de outras, não é por acaso.

Não vamos no entanto aprofundar mais esta questão, pois estaríamos a fugir ao nosso objectivo e a entrar em assuntos muito discutíveis.

Como proposta de trabalho (após apresentarmos a listagem que não nos foi possível incluir agora), procure o leitor comparar os algoritmos com as respectivas implementações.

Fácilmente se poderá aperceber que um algoritmo é uma forma muito livre e menos formal de se descrever um raciocínio, enquanto que uma função em C tem que obedecer a uma série de regras sintácticas e semânticas.

Pode ainda verificar que os algoritmos foram descritos com uma notação que se aproxima muito do C. De facto, quando os algoritmos foram elaborados, a linguagem para a respectiva implementação já tinha sido escolhida. Tal facto não elimina o carácter genérico dum algoritmo.

No próximo número será apresentada, numa forma menos extensa, a implementação de mais algumas funções do nosso programa, assim como a listagem que (por motivos técnicos) não nos foi possível incluir agora.

ALEXANDRE RODRIGUES

Leia, Divulge
e
Assine
«RS232 - Informática»

ESCOLHA DA SELECÇÃO ÓPTIMA:

- * implementação do algoritmo da escolha da selecção óptima
- * revisão dos conceitos básicos de recursividade

Depois de uma breve interrupção no desenvolvimento deste tema, vamos neste número passar à implementação do algoritmo 20 especificado no artigo anterior, que realiza a escolha da selecção óptima. Especificámos já o tipo de variáveis a utilizar e declaramos essas mesmas variáveis.

Expliquei, também, de uma forma genérica o funcionamento do presente algoritmo. Apresento de seguida a listagem do programa em BASIC ZX, que o utiliza para a realização de uma tarefa específica: a escolha de um grupo de programas, a partir de um conjunto mais vasto, que devem ser gravados numa diskette cuja capacidade é limitada, de forma a que se obtenha o lucro máximo na sua venda.

```

100 REM #####
110 REM ##
120 REM #Problema da seleccao #
130 REM #optima. #
140 REM ##
150 REM # Autoria de: #
160 REM # Alexandre Rodrigues #
170 REM ##
180 REM # © RS232-Informatica #
190 REM # Fevereiro de 1990 #
200 REM ##
210 REM #####

220 REM
230 GO TO 2140
240 REM
250 REM
260 REM *****
270 REM * Inicializa sistema *
280 REM * da STACK. *
290 REM *****

300 GO SUB 2600
310 REM *****
320 REM * Declaracao de *
330 REM * constantes. *
340 REM *****

350 LET nobj=10
360 REM *****
370 REM * Declaracao de *
380 REM * variaveis. *
390 REM *****

390 DIM o(nobj,2)
400 DIM n$(nobj,3)
410 DIM a$(nobj)
420 DIM o$(nobj)
430 LET mexp=0

```

```

440 LET o$(1)="n"
450 LET capac=140
460 RETURN
470 REM
480 REM *****
490 REM * Leitura dos dados *
500 REM * iniciais. *
510 REM *****

520 GO SUB 550
530 GO SUB 560
540 GO SUB 570
550 RETURN
560 REM
570 REM Nome dos programas
580 REM -----
590 GO SUB 600
600 PRINT AT 2,0
610 FOR J=1 TO nobj
620 INPUT ("Nome do programa "
630 :J):a$(J))a$
640 IF a$<>" " THEN
650 LET o$(J)=a$
660 PRINT J;" - ";n$(J)
670 NEXT J: PAUSE 0: RETURN
680 REM
690 REM Preço e tamanho
700 REM -----
710 GO SUB 1070
720 PRINT AT 2,0
730 LET valp=0
740 FOR J=1 TO nobj
750 INPUT ("Preço do program"
760 :a$(J):" ")a$
770 IF a$<>" " THEN
780 LET o(J,1)=VAL a$
790 IF o(J,1)<=0 OR LEN STR$
800 o(J,1)>5 THEN GO TO 700
810 INPUT ("Tamanho do progr"
820 :a$(J):" ")a$
830 IF a$<>" " THEN
840 LET o(J,2)=VAL a$
850 IF o(J,2)<=0 OR LEN STR$
860 o(J,2)>5 THEN GO TO 730
870 PRINT TAB 1;J;TAB 6;n$(J)
880 :TAB (21-LEN STR$ o(J,1))
890 :o(J,1);" ";TAB (29-LEN
900 STR$ o(J,2));o(J,2);" K"
910 LET valp=valp+o(J,1)
920 NEXT J: PAUSE 0: RETURN
930 REM
940 REM Capacidade da diskette
950 REM -----
960 CLS
970 PRINT "Capacidade actual:"
980 capac;" K"
990 INPUT "Capacidade da disk":
1000 :lette: LINE a$
1010 IF a$<>" " THEN LET capac=
1020 VAL a$
1030 IF capac=0 THEN GO TO 840

```

RECURSIVIDADE EM BASIC ZX

```

0070 RETURN
0080 REM
0090 REM *****
* Listagem de dados *
*****

0000 REM
0010 GO SUB 980
0020 GO SUB 1070
0030 GO SUB 1160
0040 RETURN
0050 REM
0060 REM Nome dos programas
0070 REM -----
0080 CLS
0090 PRINT "No. Nome"
1000 PRINT "-----"
1010 FOR J=1 TO nobj
1020 PRINT J;" - ";n$(J)
1030 NEXT J: GO SUB 2540: RETURN

1040 REM
1050 REM Preço e tamanho
1060 REM -----
1070 CLS
1080 PRINT "No. Nome Preço
Tamanho"
1090 PRINT "-----"
1100 FOR J=1 TO nobj
1110 PRINT TAB 1;J;TAB 6;n$(J)
;TAB (21-LEN STR$ o(J,1));
;o(J,1);"#";TAB (20-LEN
STR$ o(J,2));o(J,2);" K"
1120 NEXT J: GO SUB 2540: RETURN

1130 REM
1140 REM Capacidade da diskette
1150 REM -----
1160 CLS
1170 PRINT "Capacidade da diske"
; "ite:";capac
1180 PAUSE 0: RETURN
1190 REM
1200 REM *****
* Opcoes disponiveis *
*****

1210 CLS
1220 PRINT " Problema da selecc
ao optima"
1230 PRINT
1240 PRINT TAB 1;"1) Alteracao d
e dados"
1250 PRINT TAB 1;"2) Listagem de
dados"
1260 PRINT TAB 1;"3) Escolha da
seleccao optima"
1270 PRINT TAB 1;"4) Listagem da
seleccao optima"
1280 PRINT TAB 1;"5) Alteracao d
o n. de programas"
1290 PRINT TAB 1;"6) Alteracao d
o espaço total"
1300 PRINT TAB 1;"7) Ler dados d
o ficheiro"
1310 PRINT TAB 1;"8) Guardar dad
os no ficheiro"
1320 PRINT AT 20,0;"© 1990 RS232
-Informatica Ltd. by: Alexa
ndra Rodrigues"
1330 RETURN
1340 REM
1350 REM *****
* Ler opcao: out- k# *
*****

```

```

1360 LET k#=INKEY#
1370 IF k#>"0" OR k#<"1" THEN
GO TO 1360
1380 RETURN
1390 REM *****
1400 REM * Procedimento *
1410 REM * recursivo Tentativa *
1420 REM * in: *
1430 REM * i-n, do programa *
1440 REM * espt-espaço gasto *
1450 REM * valp-valor possível *
1460 REM * out: *
1470 REM * op-seleccao optima *
1480 REM * Posicoes na STACK: *
1490 REM * i - s(sp-6) *
1500 REM * espt - s(sp-5) *
1510 REM * valp - s(sp-4) *
1520 REM * Variaveis locais: *
1530 REM * i1 - s(sp-3) *
1540 REM * espt1 - s(sp-2) *
1550 REM * valp1 - s(sp-1) *
1560 REM *****

1580 LET s(sp-2)=s(sp-5)+
o(s(sp-6),2)
1590 IF s(sp-2)>capac THEN
GO TO 1780
1600 LET s#(s(sp-6))="1"
1610 IF s(sp-6)=nobj THEN
GO TO 1720
1620 LET s(sp-3)=s(sp-6)+1
1630 LET pushps(s(sp-3))
1640 GO SUB push
1650 LET pushps(s(sp-3))
1660 GO SUB push
1670 LET pushps(s(sp-6))
1680 GO SUB push
1690 LET sp=sp+3
1700 GO SUB 1580
1710 GO TO 1780
1720 IF s(sp-4)<=maxp THEN
GO TO 1750
1730 LET o#=#
1740 LET maxps(s(sp-4))
1750 LET s#(s(sp-6))=""
1760 LET s(sp-1)=s(sp-4)-
o(s(sp-6),1)
1770 IF s(sp-1)<=maxp THEN
GO TO 1910
1780 IF s(sp-6)=nobj THEN
GO TO 1890
1790 LET s(sp-3)=s(sp-6)+1
1800 LET pushps(s(sp-3))
1810 GO SUB push
1820 LET pushps(s(sp-6))
1830 GO SUB push
1840 LET pushps(s(sp-3))
1850 GO SUB push
1860 LET sp=sp+3
1870 GO SUB 1580
1880 GO TO 1910
1890 LET o#=#
1900 LET maxp=s(sp-1)
1910 LET sp=sp-6
1920 RETURN
1930 REM *****
1940 REM * Apresentacao da *
* seleccao encontrada *
*****

1950 CLS
1960 IF o$(1)="n" THEN RETURN
1970 PRINT "Programas selecciona

```

>>>>>

RECURSIVIDADE EM BASIC ZX

```

408: " "
1000 PRINT "No.   Nome   Preço
Tamanho"
1090 PRINT "-----"
2000 LET espt=0
2010 FOR J=1 TO nobj
2020 IF o$(J)<>"1" THEN GO TO
2050
2030 PRINT TAB 1;J;TAB 8;n$(J)
;TAB (21-LEN STR$ o(J,1));
;o(J,1);"#";TAB (29-LEN
STR$ o(J,2));o(J,2);" K"
2040 LET espt=espt+o(J,2)
2050 NEXT J
2060 PRINT " "
2070 PRINT "Espaco ocupado:";
espt;" K"
2080 PRINT "Espaco livre:";
capac-espt;" K"
2090 PRINT "Preco total:";maxp;
"#"
2100 GO SUB 2540
2110 RETURN
2120 REM
2130 REM *****
* Programa principal *
*****

2140 GO SUB 250
2150 GO SUB 2430
2160 GO SUB 2730
2170 GO SUB 1200
2180 GO SUB 1300
2190 GO SUB 2020
2200 GO TO 2170
2210 REM *****
* Executa opcao:in-k##
*****

2220 IF k#="1" THEN GO SUB 480:
RETURN
2230 IF k#="2" THEN GO SUB 890:
RETURN
2240 IF k#="3" THEN GO SUB 2320:
RETURN
2250 IF k#="4" THEN GO SUB 1940:
RETURN
2260 IF k#="5" THEN GO SUB 2430:
RETURN
2270 IF k#="6" THEN GO SUB 820:
RETURN
2280 IF k#="7" THEN GO SUB 2650:
RETURN
2290 IF k#="8" THEN GO SUB 2600:
RETURN
2300 REM
2310 REM *****
* Controla seleccao *
* e apresenta-a. *
*****

2320 CLS : PRINT AT 10,5; FLASH
1; BRIGHT 1;"Dados em proc"
;e esamento"
2330 DIM s$(nobj)
2340 LET maxp=0
2350 LET pushp=1: GO SUB push
2360 LET pushp=0: GO SUB push
2370 LET pushp=valp: GO SUB push
2380 LET esp=espt
2390 GO SUB 1800
2400 GO SUB 1040
2410 RETURN
2420 REM
2430 REM *****
* Alterar quantidade *
* de programas. *
*****

```

```

2440 CLS
2450 PRINT "N. de programas"
;" disponiveis:";nobj
2460 INPUT "Quer alterar ?(S/N)"
;a$
2470 IF a#<>"s" AND a#<>"S" THEN
RETURN
2480 INPUT "Numero de programas"
;" pretendidos:";nobj
2490 IF nobj<=0 THEN GO TO 2480
2500 GO SUB 300
2510 RETURN
2520 REM
2530 REM *****
* Saída na impressora *
*****

2540 PRINT : PRINT "Z - saída "
;"na impressora"
2550 IF INKEY#="" THEN GO TO
2550
2560 IF INKEY#="z" OR INKEY#="Z"
THEN COPY
2570 RETURN
2580 REM
2590 REM *****
* Guardar na diskette*
*****

2600 SAVE "objects.dat" DATA
o()n
2610 SAVE "names.dat" DATA
n#()n
2620 RETURN
2630 REM
2640 REM *****
* Ler de diskette *
*****

2650 LOAD "objects.dat" DATA
o()
2660 LOAD "names.dat" DATA
n#()
2670 LET valp=0
2680 FOR J=1 TO nobj
2690 LET valp=valp+o(J,1)
2700 NEXT J
2710 RETURN
2720 REM
2730 REM *****
* Inicializa dados *
*****

2740 CLS
2750 INPUT "Quer ler do ficheiro"
?"(S/N)";a$
2760 IF a#<>"s" AND a#<>"S" THEN
GO TO 490
2770 GO SUB 2640
2780 RETURN
2790 REM
2800 REM *****
*****
2810 REM ** Inicializar **
2820 REM ** sistema. **
2830 REM *****
*****
2840 REM
2850 DIM s(255)
2860 LET sp=1
2870 LET push=20670
2880 LET pop=30000
2890 RETURN
2900 REM
2910 REM *****
*****
2920 REM ** PUSH: **
2930 REM ** in: pushp **
2940 REM ** out: sp **
2950 REM *****
*****

```

>>>>>

```

0060 REM
0070 LET S(SP)=PUSHP
0080 LET SP=SP+1
0090 IF SP=257 THEN LET SP=1
0000 RETURN
0010 REM
0020 REM *****
0030 REM ** POP: **
0040 REM ** in: - **
0050 REM ** out: POPP **
0060 REM *****
0070 REM
0080 LET SP=SP-1
0090 IF SP=0 THEN LET SP=256
0100 LET POPP=S(SP)
0110 RETURN
    
```

Este programa é constituído por várias subrotinas que podem ser divididas em quatro grupos distintos:

1)- Subrotinas de leitura de dados

Estas subrotinas realizam a leitura de toda a informação, que é lida a partir da consola do computador, ou de um ficheiro em diskette.

Linha	Descrição
560	leitura do nome dos programas existentes;
660	leitura do preço e tamanho dos programas existentes;
820	leitura da capacidade da diskette (em Kbytes);
1350	leitura da opção a ser executada;
2650	leitura da dados já definidos em ficheiro;

2)- Subrotinas de saída de dados

Estas subrotinas realizam a saída de toda a informação para o ecran, ou para um ficheiro em diskette.

Linha	Descrição
980	listagem do nome dos programas existentes;
1070	listagem do preço e tamanho dos programas existentes;
1160	impressão da capacidade da diskette no ecran;
1200	listagem das opções que se podem realizar;
1950	lista a solução encontrada;
2600	escrita dos dados em ficheiro;

3)- Subrotinas de controle

Estas subrotinas controlam a execução do programa, em função das atitudes tomadas pelo utilizador. São, por isso, subrotinas de interacção com o utilizador, que realizam breves acções de entrada/saída e invocam frequentemente as subrotinas de entrada e saída de dados, bem como as subrotinas de processamento de informação.

Linha	Descrição
2140	controle de funcionamento do programa;
2220	controle da execução das opções existentes;
2320	controle da construção da solução óptima; esta subrotina invoca a subrotina que implementa o al-

goritmo 20, passando os parâmetros de entrada e alocando espaço para as variáveis locais na stack;
 2430 controle da alteração do número de programas existentes;
 2530 controle da saída de dados na impressora;
 2730 controle da inicialização dos dados;

4)- Subrotinas de processamento de informação

Estas subrotinas são geralmente as mais importantes dentro de um programa. São estas subrotinas que "pegam" na informação fornecida pelo utilizador e "transformam-na" noutra informação - ou seja, "produzem" nova informação em função da primeira. Note-se que o conceito de informação é, neste ponto, crucial. Num computador a informação é implementada sob a forma de dados que se encontram armazenados em memória (variáveis, parâmetros, constantes, etc.).

De facto, são estas subrotinas que realizam a tarefa pretendida pelo utilizador, embora não o pudessem fazer sem a existência das anteriores.

Linha	Descrição
1580	processamento da solução do problema; esta subrotina implementa o algoritmo 20 que faz a construção da selecção óptima; trata-se da principal subrotina de todo o programa;
2800	subrotina que inicializa o sistema que simula a stack;
2970	implementação da operação PUSH na stack;
3080	implementação da operação POP na stack;

5)- Subrotinas de inicialização do programa

Estas subrotinas encarregam-se de criar as condições necessárias ao funcionamento do programa. No presente caso, entre outras tarefas, é necessária a declaração das variáveis e constantes a utilizar dada a linguagem que estamos a utilizar.

Linha	Descrição
260	inicialização do sistema da stack;
300	declaração de constantes;
360	declaração e inicialização de variáveis globais.

É muito importante que o leitor se aperceba que a divisão de subrotinas em diferentes classes é essencial para a programação bem estruturada. Quando se trabalha com linguagens pouco estruturadas, como é o caso do BASIC ZX, mais claro ainda se torna este facto. De resto, a divisão feita para este programa pode ser aplicada a qualquer outro, uma vez que as leis de estruturação de um programa são genéricas.

Vamos agora analisar o funcionamento deste programa. Se fizermos RUN é iniciada a sua execução na subrotina de controle da linha 2140. Esta subrotina pode ser descrita algorítmicamente de uma forma muito simples:

>>>>>>

ELECTRÓNICA 45

Em colaboração com Commodore Portuguesa Electrónica, S.A.
apresenta:

AMIGA 3000

Venha conhecê-lo !!!

Temos soluções completas (software e hardware) para as suas necessidades
com equipamentos Commodore AMIGA

**MARQUE UMA DEMONSTRAÇÃO E EXPONHA-NOS O SEU CASO !
FAZEMOS DEMONSTRAÇÕES EM QUALQUER ÀREA ESPECÍFICA !**

Preços Especiais e facilidades de pagamento !

HARDWARE:

- Toda a gama Amiga: 500/2000/2500/3000
- Placa de emulação XT par A500
- Placas de emulação XT/AT para A2000 e A3000
- Digitalizadores de imagem e som
- Genlocks
- MIDI

- Expansões de memória p/A500/A2000/A3000
- Drives 3.5"(880kb/1,76M),5.25(360kb/880Kb)
- HardDisks p/A500/Hardcards para A2000

Software Amiga:

- Importamos software original
- Demonstrações de software

Software MULTISOFT para PC:

- Software nacional de gestão com assistência e garantia.
- Área médica - Cirurgia Plástica, Oftalmologia
- Área de Gestão - Contabilidade, Salários, Stocks Facturação, Encomendas, Compras, Gestão de Bancos, Gestão de Caixa.
- Área Ourivesaria/JOalharia: Joia, Jóia (Ourivesaria).

CURSOS DE FORMAÇÃO COMMODORE AMIGA
Já começaram !!! INFORME - SE JÁ

Centro Comercial Torre das Flores, Loja-16 - 2795 Linda-a-Velha __ Em frente ao terminal da Carris, Autocarros 48/51 - Tel:4198759

Aberto 7 dias por semana, das 10.30 às 23.00 (encerramento para almoço e jantar)

RECURSIVIDADE EM BASIC ZX

Algoritmo21 : (controle principal da execução do programa)

```
[ Inicializa programa ]
[ Permite alteração do nº de programas existentes ]
[ Inicializa dados - leitura do teclado ou do ficheiro ]
Repete
  [ Imprime opções disponíveis ]
  [ Lê opção pretendida do teclado ]
  [ Executa a opção escolhida ]
Até [ sempre ]
```

Todas as subrotinas que constituem o programa podem ser descritas de uma forma algorítmica, mas como já referi nem todas justificam tal tarefa.

Assim, o programa pergunta-nos se pretendemos alterar o número de programas existentes. Vamos pois responder que não, ficando esta variável (nobj) com o valor 10, com o qual é inicializada.

Seguidamente o programa pergunta se é pretendida a leitura de dados de um ficheiro. Como não temos à partida nenhum ficheiro com dados vamos novamente responder que não; é-nos então pedida a introdução do nome dos programas, e seguidamente dos respectivos preços. Introduza o leitor os dados de acordo com a seguinte tabela:

No.	Nome	Preço	Tamanho
100	Keops	10#	10 K
101	Kdops	11#	11 K
102	Keops	12#	12 K
103	Kdops	13#	13 K
104	Keops	14#	14 K
105	Kdops	15#	15 K
106	Keops	16#	16 K
107	Kdops	17#	17 K
108	Keops	18#	18 K
109	Kdops	19#	19 K
110	Keops	20#	20 K

Z - saída na impressora

Após a introdução destes dados, é apresentada a lista de opções que se podem executar. Vamos agora pôr à prova a eficiência do nosso programa, ensaiando para isso várias hipóteses.

Começamos por escolher a opção 6, que nos permite alterar a capacidade da diskette que irá conter os programas; vamos então atribuir-lhe a capacidade de 10 Kbytes. Podemos agora pedir ao computador que nos diga qual a solução óptima, através da terceira opção da lista.

O resultado obtido é o seguinte:

```
Programas seleccionados:
No. Nome Preço Tamanho
1 Keops 10# 10 K

Espaco ocupado:10 K
Espaco livre:10 K
Preço total:10#

Z - saída na impressora
```

Se olharmos para a tabela anterior, é óbvio que a melhor solução é escolher apenas o programa 1 (Keops), visto que é o único que satisfaz a capacidade da diskette. Mais nenhum programa poderia ser escolhido. Esta situação, embora muito simples e fácil de resolver, mostrou-nos que o programa fez o pretendido.

Vamos novamente alterar a capacidade da nossa diskette para 20 Kbytes, através da opção 6.

Pedimos novamente ao computador que encontre a melhor solução (opção 3), e a resposta é a seguinte:

>>>>>

J. P. R. Informática

DISKETTES 3.5 DD

(PREÇOS ESPECIAIS PARA REVENDA)

SONY ; MITSUBISHI ; TDK ; KAO "COLOR" ; 3M ; VERBANTIM ; MAXELL
POLAROID ; BASF ;

(Outras diskettes sem marca, às côres)

Revenda de Software

R. GENERAL ALVES ROÇADAS 38 - 40 ; Loja 10 Centro C. ACSANTOS
2675 ODIVELAS

Telefone / Fax: 01 - 987 72 91

RECURSIVIDADE EM BASIC ZX

Programas seleccionados:

No.	Nome	Preço	Tamanho
7	Doomdark	27\$	16 K

Espaco ocupado: 16 K
Espaco livre: 4 K
Preço total: 27\$

Z - saída na impressora

Novamente podemos verificar que a melhor solução consiste na escolha do programa 7, uma vez que só cabe um programa na diskette e qualquer um destes pode ser escolhido. O programa 7 é aquele que vale mais dinheiro e por isso foi escolhido de entre os dez possíveis. Esta situação é ainda muito simples, mas permite-nos verificar facilmente que a solução apresentada pelo computador é a melhor, como se pretende.

Aumentemos agora a capacidade da diskette para 30 Kbytes; o resultado obtido é o seguinte:

Programas seleccionados:

No.	Nome	Preço	Tamanho
5	Fist	25\$	14 K
7	Doomdark	27\$	16 K

Espaco ocupado: 30 K
Espaco livre: 0 K
Preço total: 52\$

Z - saída na impressora

Desta vez não é tão óbvio que a solução apresentada é a melhor. No entanto, podemos verificar de imediato que os programas ocupam toda a diskette e dão um lucro de 52\$00. Existem mais hipóteses, como por exemplo a escolha dos programas 4 e 8, que juntos também ocupam toda a diskette mas dão um lucro apenas de 42\$00, inferior ao da solução apresentada pelo computador. De resto, podemos tentar qualquer combinação possível de programas que caiba na diskette, que não conseguimos obter nenhuma cujo lucro em dinheiro seja superior a 52\$00.

De facto o computador ensaiou todas essas hipóteses pelo processo descrito no algoritmo 20, e conclui que a solução óptima é a que nos apresentou; **NÃO EXISTE NENHUMA SOLUÇÃO MELHOR.**

Se quisermos pôr esta realidade à prova, podemos tentar situações mais complicadas alterando novamente a capacidade da diskette para valores mais elevados, Para uma capacidade de 90 Kbytes o computador construiu a seguinte selecção de programas:

Programas seleccionados:

No.	Nome	Preço	Tamanho
1	Keops	18\$	10 K
2	Mday3	20\$	11 K
4	Karate2	19\$	13 K
5	Fist	25\$	14 K
7	sei comb	21\$	15 K
7	Doomdark	27\$	16 K
8	Midnight	23\$	17 K
9	Rings	20\$	18 K
10	Boxe	24\$	19 K

Espaco ocupado: 90 K
Espaco livre: 0 K
Preço total: 139\$

Z - saída na impressora

Nenhum outro conjunto de programas que caiba na diskette dá um lucro superior a 139\$00. Novamente esta é a selecção óptima. Para uma capacidade da diskette de 120 Kbyte, a solução óptima é a seguinte:

Programas seleccionados:

No.	Nome	Preço	Tamanho
1	Keops	18\$	10 K
2	Mday3	20\$	11 K
4	Karate2	19\$	13 K
5	Fist	25\$	14 K
7	sei comb	21\$	15 K
7	Doomdark	27\$	16 K
8	Midnight	23\$	17 K
9	Rings	20\$	18 K
10	Boxe	24\$	19 K

Espaco ocupado: 120 K
Espaco livre: 0 K
Preço total: 183\$

Z - saída na impressora

Se na realidade quiséssemos utilizar uma diskette compatível com o TIMEX FDD 3000, cuja capacidade em cada lado é de 140 Kbyte, a solução obtida seria a seguinte:

Programas seleccionados:

No.	Nome	Preço	Tamanho
1	Keops	18\$	10 K
2	Mday3	20\$	11 K
4	Karate2	19\$	13 K
5	Fist	25\$	14 K
7	sei comb	21\$	15 K
7	Doomdark	27\$	16 K
8	Midnight	23\$	17 K
9	Rings	20\$	18 K
10	Boxe	24\$	19 K

Espaco ocupado: 133 K
Espaco livre: 7 K
Preço total: 202\$

Z - saída na impressora

Muitas outras situações poderiam ser ensaiadas, através da alteração do número de programas, dos seus preços e dos seus tamanhos. O leitor pode agora explorar o funcionamento deste programa através das várias opções disponíveis.

Muito mais importante do que testar o funcionamento do programa, será dele tirar o melhor proveito. Não se pense no entanto que a sua única utilidade é a de escolher a melhor selecção de um conjunto de programas a gravar numa diskette. De facto, este foi apenas um dos inúmeros casos em que um programa deste tipo pode ser utilizado; eis alguns dos exemplos mais vulgares:

- * escolha do conjunto de mercadorias a colocar num navio sabendo que o navio tem um limite para o peso da carga que transporta;
- * escolha das peças de vestuário que se pretendem fabricar a partir de um dado tecido, cuja área é limitada;
- * escolha do conjunto de máquinas que devem ser adquiridas por uma fábrica, sabendo-se que essas máquinas têm tamanhos diferentes, preços de compra diferentes, rendimento (número de fabricados por dia) diferentes, e que o pavilhão que irá conter essas máquinas tem uma área limitada;
- * escolha dos automóveis que se pretendem pôr em exposição num stand, para venda, sabendo que esse stand tem uma área limitada.

Estes exemplos chamam-nos de imediato a atenção para o facto de introduzirem elementos novos em relação ao problema que acabámos de estudar. Assim, no primeiro caso podemos verificar pela forma como o problema é enunciado, que se trata de um caso que pode ser resolvido pelo nosso programa, bastando para isso fazer a substituição dos seguintes termos:

programa ————— > mercadoria
 tamanho em Kbyte ——— > peso em Kg
 capacidade da diskette → carga máxima

O nosso programa não necessita de qualquer alteração. Contudo, se formos um pouco mais realistas podemos objectar que, por um lado, as mercadorias além de terem um certo peso ocupam também um determinado espaço; por outro lado, o navio tem um espaço limitado para armazenar as mercadorias a transportar. Esta é uma situação que o nosso programa não contempla. Quais as diferenças introduzidas por esta nova situação?

A diferença fundamental consiste no facto dos elementos a serem escolhidos para a selecção óptima passarem a ter mais um atributo que limita a construção dessa selecção.

Esse atributo é o espaço que uma mercadoria ocupa. Consequentemente, passa a existir mais uma condição para que uma solução seja aceite; essa condição consiste no facto de todas as mercadorias que constituem essa solução não excederem o espaço disponível no navio.

Feitas estas observações, quais as alterações a fazer ao nosso programa? Antes de respondermos a esta questão, devemos pensar primeiro nas alterações que devem ser feitas ao algoritmo 20. Essas alterações não são muito profundas, uma vez que o raciocínio a aplicar é exactamente o mesmo.

O algoritmo 18 que foi apresentado na revista Nº 22 (e que voltarei a repetir mais adiante), descreve um raciocínio totalmente genérico, e é dele que devemos partir para resolvermos este novo problema. As diferenças irão incidir essencialmente na implementação dos predicados [inclusão aceitável] e [exclusão aceitável].

Vou deixar para mais tarde esta questão, mas proponho ao leitor que se debruce sobre ela.

Vamos agora examinar o segundo caso. Aqui o problema principal levanta-se pelo facto de uma peça de vestuário, além de ocupar um determinado espaço, ter uma determinada forma. Novamente, os elementos a serem escolhidos passam a ter mais um atributo que limita a construção da solução.

Para que seja possível recortar uma peça de tecido, além de ter que haver o espaço suficiente é também essencial que esse mesmo espaço tenha uma forma compatível com a forma da peça. Trata-se pois de uma situação bastante semelhante à anterior, com a diferença de que o tecido continua a ter apenas um atributo (a sua área), enquanto que o porão do navio passou a ter dois atributos: carga máxima e área disponível.

As diferenças a serem introduzidas no programa, são por isso ligeiramente diferentes.

O terceiro caso introduz mais uma novidade. Quanto aos atributos dos elementos a serem escolhidos, passam a ser três: o espaço que a máquina ocupa, o seu preço e o seu rendimento. A única limitação que temos para a construção de uma solução consiste no espaço disponível na sala das máquinas, limitação essa que está relacionada com o primeiro atributo dos elementos (espaço que a máquina ocupa). Quanto aos outros dois atributos (preço e rendimento), eles vão exercer a sua influência na decisão de uma determinada solução ser melhor do que outra - pretende-se um rendimento máximo e um preço mínimo.

Enquanto que nos dois casos anteriores tínhamos dois atributos a limitar a construção de uma solução, e um atributo a influenciar na descrição de escolha de "melhor" solução, neste caso temos apenas um atributo a limitar a construção de uma solução, e dois atributos a influenciar a escolha da "melhor" solução. Assim sendo, as diferenças na implementação do algoritmo 18 vão incidir no significado da acção [verifica se a solução é óptima].

Finalmente, o quarto caso introduz uma novidade curiosa.

Quando, ao explicar a certa pessoa que possui um stand de venda de automóveis, qual a utilidade de um algoritmo de backtracking de escolha de solução óptima, coloquei-lhe a seguinte questão:

"suponha que pretende saber quais os automóveis que deve ter em exposição, de forma a que tenha um lucro máximo nas vendas". Expliquei-lhe então que o factor que limitava essa escolha era o espaço disponível no stand, que não lhe permitia colocar em exposição todos os automóveis disponíveis. Havia então que fazer uma escolha, baseada no tamanho dos automóveis, nos seus lucros de venda, e na área do stand. Esse problema poderia ser facilmente resolvido pelo nosso programa. Existe no entanto o problema da forma que os carros têm, como acontecia com as peças que se pretendem recortar de

tecido. Já abordámos pois, esse problema.

Essa mesma pessoa levantou-me uma questão muito interessante quando perguntou: " e se a solução óptima só incluir carros de uma marca? também me interessa ter alguma variedade de marcas na minha exposição! "

Este problema não acrescenta qualquer atributo aos elementos a serem seleccionados, como acontece nos casos anteriores; no entanto faz-nos concluir que uma solução não é óptima apenas por causa do lucro que se obtém na venda, mas também o é em função da variedade de marcas que contém.

Assim, somos obrigados a acrescentar mais um atributo aos elementos em causa: a marca do carro.

Este novo atributo não limita a construção de uma solução, mas vai exercer influência na decisão de "melhor" solução.

Desta forma, as diferenças a serem introduzidas na implementação do algoritmo 18 vão incidir no significado da acção [verifica se a solução é óptima], tal como acontecia no caso anterior.

Proponho ao leitor que se debruce sobre estes quatro casos e tente identificar quais as diferenças a serem incluídas na implementação do algoritmo 18, que passo a descrever:

Algoritmo 18: (procura da solução óptima)

Tentativa (in:i)

Se inclusão aceitável] então

[incluir i-ésimo elemento]

Se $i < n$ então

Tentativa (i+1)

senão

[verifica se a solução é óptima]

fimse

Se [exclusão aceitável] então

Se $i < n$ então

Tentativa (i + 1)

senão

[verifica se a solução é óptima]

fimse

fimse

Podemos ir mais longe e pensar em implementar este algoritmo de forma a que qualquer dos casos atrás referidos seria solucionado pelo programa; obtemos dessa forma um programa completamente genérico. No próximo artigo irei explicar as alterações que se devem fazer na implementação deste algoritmo e apresentarei essa mesma implementação. Iremos ainda ver que outros algoritmos são possíveis para a escolha de uma "boa" solução, que nem sempre é a solução óptima. É deveras engraçado ver os algoritmos "competir" entre si em termos de velocidade de resposta e em termos de solução obtida. Iremos também iniciar uma revisão mais completa e científica ao conceito de recursividade e sua utilização. **ATÉ BREVE.**

ALEXANDRE RODRIGUES



SANCHES GIRÃO

REPARAÇÕES

Computadores

TV
VIDEO
Hi - Fi

ANTENAS
Satélite
Terrestres

Rua Edith Cável, 7 1900 Lisboa

Tel: 82 24 81

MSX - Gráficos & Som PSG

Gráficos Avançados (I)

CARACTERÍSTICAS DE CADA MODO DE ECRAN.

O MSX tem um dos mais versáteis chips gráficos, o TMS 9929 A, desenvolvido pela Texas Instruments, Inc. dos Estados Unidos. É um dos mais compreensivos chips gráficos disponíveis e que fornece uma grande variedade de facilidades, como gráficos de alta resolução em 16 cores e animação de sprites. Ele pode até gerir a sua própria RAM de 16K, o que significa que o processador central não precisa de fornecer memória para o gráfico no modo de tela.

O MSX-BASIC foi escrito especialmente com essas fortes capacidades gráficas e facilidades de programação para iniciados. É fácil de ser usado, uma vez experimentado, e pode produzir algumas figuras espectaculares se se tentar com afinco. Vamos então iniciar explicando em primeiro lugar os modos de exibição disponíveis no MSX.

Modo	Texto/Gráfico	Resolução	Cores	Entrada	Gráficos	Sprites
0	40 x 24 modo texto	40 x 24 car	2 em 16	sim	não	não
1	32x24 modo	32x24 car	2 em 16	sim	não	sim
2	gráf. alta res.	256x192 caracteres	16	não	sim	sim
3	Multicores	64x68 blocos	16	não	sim	sim

Notas:

ENTRADA: uso ou não de comando INPUT no modo de tela dado. Não podemos fazer entrada enquanto estiver em um modo gráfico.

GRÁFICOS: uso de comandos gráficos, por exemplo DRAW, no modo de ecran dado. Geralmente não podemos usar comandos gráficos em modo de texto (0 e 1).

Comandos associados a TODOS os modos de ecran:

SCREEN Define modo de ecran.
CLS Limpa o ecran.
COLOR Define as cores do texto, do fundo e das margens

Comandos e funções associados SOMENTE a modo texto:

WIDTH Define a largura do modo de ecran do texto.

LOCATE Define a posição do cursor no modo de ecran do texto
TAB Define a posição horizontal do cursor na linha actual.
CSRLIN Retorna à posição vertical actual do cursor.
POS (0) Retorna à posição horizontal actual do cursor.

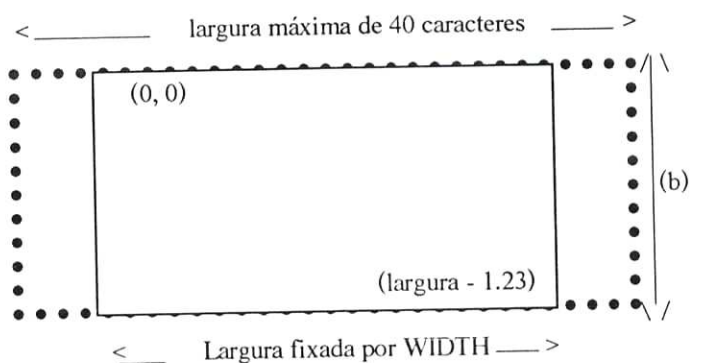
Comandos e funções associados SOMENTE a modos gráficos

CIRCLE Desenha um circulo.
DRAW Desenha de acordo com a macrolinguagem (a)
LINE Desenha linhas, quadrados e rectângulos.
PAINT Pinta com a cor de texto actual.
PSET Desenha um ponto.
PRESET Elimina um ponto.
POINT Retorna à cor do ponto referido.

Comandos associados ao Processador de Exibição do Video (VDP):

VPOKE Executa uma sentença POKE na RAM de video.
VPEEK Executa uma sentença PEEK na RAM de video.
VDP Retorna valores do registador VDP.
BASE Retorna o endereço base das tabelas das RAM de video

MODO 0: MODO DE TEXTO 40 x 24



Atenção Leitores - MSX

Na edição de Setembro, faltou incluir uma linha do programa nela apresentado:
 4360 DATA1, 130, 20, 64, 10, 160, 82, 0

MSX - Gráficos & Som PSG

O MODO 0 dá-nos 40 caracteres por linha, que é o número máximo de caracteres que se pode ter numa linha, em computadores MSX. Nesse modo o leitor pode escrever e editar o seu programa. Descobrirá que a exibição de um programa neste modo é mais fácil de se ler.

Entretanto, o MODO 0 tem algumas desvantagens. Por exemplo, os caracteres são exibidos no formato comprimido, isto é, 6 x 8 em vez de 8 x 8 e, dessa forma, alguns caracteres gráficos irão aparecer cortados. Os dois pontos mais à direita em cada caracter não são exibidos! Entretanto, isto não afecta caracteres alfanuméricos.

A largura padrão do modo de ecran no MODO 0 é de 37 caracteres. No entanto pode-se aumentar a largura de exibição para 40 caracteres, usando o comando WIDTH. As coordenadas do canto superior esquerdo são (0, 0), enquanto que o canto inferior direito é (WIDTH 1, 23), ou quando inicializado é (38, 32). Pode ainda posicionar o cursor de texto usando TAB e LOCATE. Para descobrir onde está o cursor, são usadas as funções POS (0) e CSRLIN.

No MODO 0, não pode usar SPRITES e também só pode usar duas das 16 cores, apesar do facto da combinação de duas cores ser da sua livre escolha. As cores padrões são as mesmas do MODO 1 : texto em branco e fundo azul.

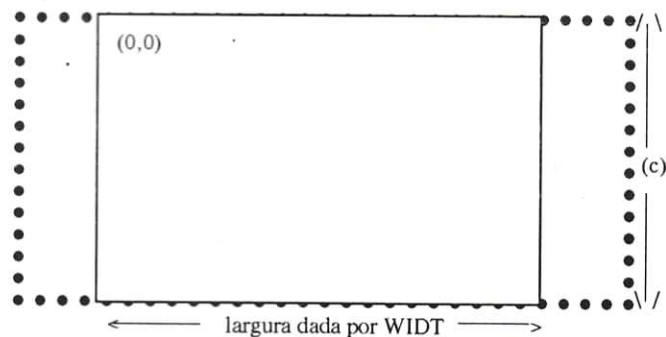
Observe que o MODO 0 não tem qualquer margem: ela simplesmente desaparece da exibição, de modo que definir cores no MODO 0 é completamente inútil. Observe que as cores da exibição irão mudar imediatamente após a execução do comando COLOR.

Apesar de ser modo gráfico, o leitor está livre de usar comandos INPUT e a lista das teclas de funções é exibida na linha 23, a menos que seja desactivada por KEY OFF.

Todos os comandos gráficos e funções deste modo serão tratados como "Illegal function call", de modo que vamos examiná-las.

MODO 1 : MODO TEXTO 32 x 24

← largura máxima de 32 caracteres →



(c) 24 caracteres

O MODO 1 tem uma resolução de 32 x 24 caracteres, mas não é gráfico. Neste modo podem-se escrever e editar programas. Ele pode exibir 8 x 8 caracteres sem qualquer corte (como no caso do MODO 0).

Pode-se aumentar a largura de exibição para 32 caracteres usando o comando WIDTH. O motivo para a largura padrão ser menor que 32 é que algumas TVs e monitores não podem exibir o modo de ecran inteiro.

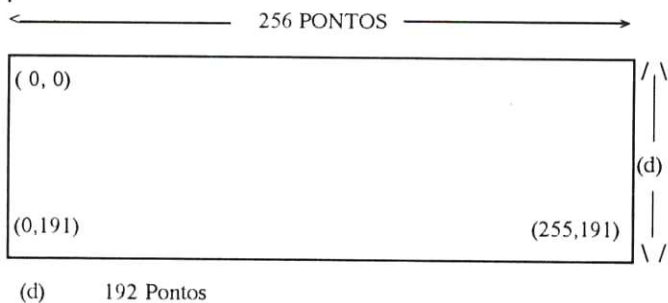
A coordenada do canto superior esquerdo é (0, 0), enquanto que a do canto inferior direito é (WIDTH- 1,23).

Pode posicionar o cursor de texto usando TAB e LOCATE. Para descobrir onde está o cursor, use as funções POS(0) e CSRLIN. Podem-se usar somente 2 das 16 cores, apesar delas serem da sua inteira escolha. Observe que as cores de exibição mudam imediatamente após a execução do comando COLOR.

Apesar de ser modo gráfico, não existe qualquer problema em se usar o comando INPUT, e a lista das teclas de função é exibida na linha 23 a menos que seja desactivada por KEY OFF.

Todos os comandos gráficos e funções neste modo, excepto aquelas envolvidas com sprites, serão tratadas como chamadas ilegais de função.

MODO 2: Modo Gráfico de Alta Resolução de 256 x 192:



O MODO 2 é o modo gráfico mais largamente usado, pois fornece ao utilizador um modo gráfico de ecran de alta resolução com 16 cores.

Neste modo podem-se usar sprites e a microlinguagem gráfica através do comando DRAW. este modo é usado na maioria dos programas de jogos.

A sua resolução horizontal é de 256 pontos, enquanto que a resolução vertical é de 192. Entretanto, a resolução de cores é um tanto diferente: é de 32 x 192. Isto significa que somente se podem seleccionar duas cores por bloco horizontal de 8 x 1 ponto. Pode seleccionar a cor de fundo e a de texto para cada bloco de 8 pontos, mas se desenhar um ponto com uma terceira cor, os pontos desenhados na cor anterior dentro desse bloco irão alterar-se para a nova cor de fundo, automaticamente.

Devido a isso, podem-se obter figuras "borradas" se não se for cauteloso com o local onde está desenhado e com que cor. Entretanto, se você desenhar cuidadosamente com a resolu-

MSX - Gráficos & Som PSG

ção de 8 cores, poderá produzir belas figuras.

Podem-se usar sprites neste modo.

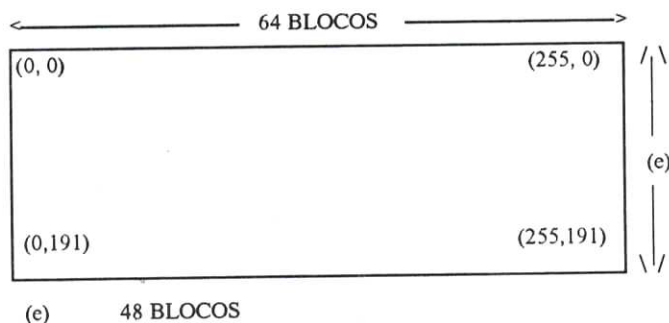
Os sprites estão num plano do modo de ecran diferente do plano de ecran principal. Isto significa que não importa onde os sprites sejam exibidos; eles não afectam o que é desenhado no modo de ecran principal, de modo nenhum.

Mais adiante iremos ver que quando um sprite ocupa uma parte do modo de ecran em particular, o modo de ecran principal continua totalmente inalterado. No modo gráfico 2, a lista das teclas de função não pode ser exibida. Irá descobrir também que o comando PRINT não funciona neste modo. Para imprimir no modo gráfico de ecran deve-se abrir um arquivo no modo de ecran usando PRINT#.

O comando COLOR não altera as cores imediatamente neste modo. Para alterar as cores de um modo de ecran inteiro deve-se executar um comando CLS.

Não se pode utilizar o comando INPUT neste modo, porque ele força o modo de ecran a retornar para o modo de texto usado anteriormente.

MODO 3: Gráficos de Baixa Resolução Multicores de 64x48



O MODO 3, Multicolorido, dá uma resolução baixa de 64 x 48 blocos, mas os blocos individuais podem ter as suas próprias cores (ao contrário do MODO 2). O leitor não irá obter efeitos de borrões como no MODO 2.

As suas coordenadas são as mesmas do MODO 2; mas grupos de 4 por 4 pontos, representam um bloco. Isto permite que se desenhe nas mesmas coordenadas como no MODO 2, o que é vantajoso.

Podem-se usar sprites neste modo e da mesma maneira que no MODO 2.

No MODO 3, a lista das teclas de função não é exibida. Também descobrirá que o comando PRINT não funciona neste modo. Para imprimir no modo gráfico de ecran, deve-se primeiro abrir um arquivo no modo de ecran e usar PRINT#.

O comando COLOR não altera a cor imediatamente, neste modo. Para alterar a cor do modo de ecran inteiro, deve-se usar o comando CLS.

Neste modo também não se pode usar o comando INPUT pois iria forçar o modo de ecran de volta para o modo usado anteriormente.

Continuaremos no próximo número!!!

Até lá, faça experiências... ok?

OSVALDO DUARTE

EXECUTAMOS

PROCESSAMENTO DE TEXTO

Folhetos Publicitários ; Maquetes de anúncios

EM SISTEMA DE EDIÇÃO ELECTRÓNICA

Calçada de Palma de Baixo, 15 - cv Drr^o 1600 Lisboa Tel: 726 46 52



Commodore

SOFTCLUB



Commodore

COMMODORE 64
AMIGA 500 / 2000 / 3000
ATARI ST
PC 10 III / 20 / 30 / 40

MONITORES : COMMODORE ; PHILIPS ; NEC
IMPRESSORAS : EPSON ; NEC

MINI-GENLOCK; RENDALE 8852; DISIVIEW GOLD
MIDI INTERFACE E SAMPLERS.

Tudo para o seu **AMIGA**

(Enviamos Software à cobrança)

Centro Comercial City, loja 18 - 2º Piso * R.Tomás Ribeiro, 34 A/B 1000 Lisboa TeL:3528452

PRECISA-SE

VENDEDOR DE INFORMÁTICA M / F

PRETENDEMOS UM VENDEDOR DE EQUIPAMENTO INFORMÁTICO
A FIM DE FAZER FACE À EXPANSÃO DOS NEGÓCIOS COM SOLUÇÕES
COMPLETAS:

FUNÇÃO: Comercialização de computadores pessoais, periféricos e outros equipamentos informáticos.

PERFIL:

- * Habilitações mínimas: 11º ano
- * Idade entre 23 e 35 anos
- * Bons conhecimentos de Inglês
- * Carta de Condução
- * Facilidade em estabelecer e desenvolver contactos
- * Experiência de vendas

OFERECE-SE:

- * Remuneração adequada ao nível da função e experiência do candidato
- * Regalias Sociais
- * Formação para desenvolvimento da carreira

Resposta com curriculum vitae detalhado para: APARTADO 210 2745 QUELUZ

A MATEMÁTICA NO SEU COMPUTADOR

Desde a sua introdução como operador matemático pelo célebre Pierre Fermat (1601 - 1665), a derivação tem feito marcar a sua presença em numerosíssimos ramos da matemática pura e aplicada tendo-se mesmo tornado no conceito central de um desses ramos: o Cálculo diferencial.

Pode-se, pois, concluir que o cálculo da derivada de uma função seja uma tarefa corrente na resolução de vários problemas e que, portanto, a sua computação assuma um interesse especial. Hoje em dia, existem programas comerciais (como o "Derive" ou o "Mathematica") que efectuam derivação simbólica; porém, em grande parte dos casos, só nos interessa o valor numérico da derivada em determinados pontos, tarefa que pode ser levada a cabo de uma forma bem mais simples. Embora a derivação numérica seja um problema delicado em virtude de não ser possível garantir a proximidade das derivadas da função e do polinómio interpolador usado para aproximá-la, o seguinte par de pequenas mas úteis rotinas servirá, na maior parte dos casos, para obter resultados suficientemente satisfatórios.

As duas rotinas só abordam, respectivamente, o cálculo das primeira e segunda derivada visto serem as que aparecem mais frequentemente na prática.

```

10 REM  Derivada de 1ª ordem
      Por  Joao Fraga
20 CLS
30 INPUT "F(x)=";a$: PRINT "F(
x)=";a$
40 INPUT "X0=";x1: PRINT "X0="
;x1
50 DEF FN f(x)=VAL a$
60 LET x0=x1-1/2+8: LET x2=x1+
1/2+8
70 LET y0=FN f(x0): LET y1=FN
f(x1): LET y2=FN f(x2)
80 LET d1=(x1-x2)+y0/(x0-x1)/(
x0-x2)
90 LET d2=(2*x1-x0-x2)+y1/(x1-
x0)/(x1-x2)
100 LET d3=(2*x1-x0-x1)+y2/(x2-
x0)/(x2-x1)
110 PRINT "Derivada=";d1+d2+d3
120 PRINT : GO TO 30

```

```

10 REM  Derivada de 2ª ordem
      Por  Joao Fraga
20 CLS
30 INPUT "F(x)=";a$: PRINT "F(
x)=";a$
40 INPUT "X0=";x1: PRINT "X0="
;x1
50 DEF FN f(x)=VAL a$
60 LET x0=x1-1/2+8: LET x2=x1+
1/2+8
70 LET y0=FN f(x0): LET y1=FN
f(x1): LET y2=FN f(x2)
80 LET u=y0/(x0-x1)/(x0-x2)+y1
/(x1-x0)/(x1-x2)+y2/(x2-x0)/(x2-
x1)
90 PRINT "Derivada=";2*u
100 PRINT : GO TO 30

```

VENDO (e troco)

Software Utilitário, em diskette
para SPECTRUM PLUS 3

Informações:

JOSÉ CASANOVA

R. GEN. SILVEIRA, 55 - 1º 4000 PORTO

ABOUT MIDI,...???

Com este artigo, vamos dar início a um dossier de acesso e domínio público sobre o Protocolo MIDI.

Introdução:

Facto 1- Durante a década de 70, com a vulgarização dos processadores e conseqüentemente dos micro computadores, sentiu-se a necessidade de comunicar, bem e rapidamente entre máquinas diferentes a grandes velocidades.

Assim, tornaram-se comuns os termos RS232, RS242, IEEE....

Facto 2- Facilmente se conclui que a música estava fechada numa ampola em que muito pouca gente sem a necessária preparação, tinha hipótese de ser integrada.

Assim, com a descida dos preços dos materiais e equipamentos informáticos, surge uma luz capaz de abalar a pequena ampola...

Facto 3- Com a proliferação de multiplas linguagens mais ou menos académicas, mais ou menos profissionais, a inexistência de padrões internacionais de programação reconhecidos e adoptados, tornou-se imperioso a existência de algo externo aos equipamentos que fosse comum a todos eles:

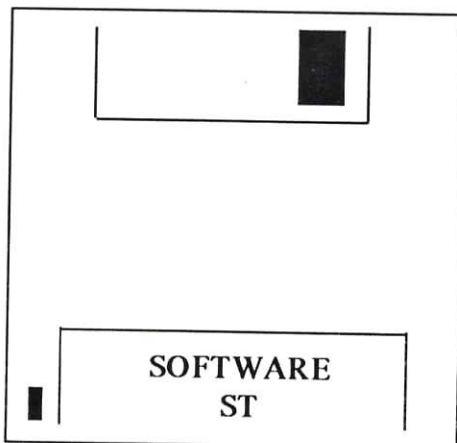
- A comunicação.

Facto 4- Quem tem conhecimentos do meio musical, verificará que não estamos muito longe da verdade, quando afirmamos que o preço de uma hora-estúdio era durante a década passada e na de 70, quase o dobro do preço actual, deduzidos que estejam, é lógico, os efeitos inflacionários correspondentes.

(Continua)

VICTOR FRANCISCO

MALHUS Informática



ATARI ST E (UK Version)

Na compra de um ATARI ST oferecemos software !!!

Apresente-nos o seu problema !
CONSULTE - NOS !!!

Outras marcas:

AMSTRAD - COMMODORE - EPSON - HYUNDAI - PHILIPS - SCHNEIDER - SINCLAIR

*Uma vasta gama de periféricos profissionais !
Serviços de assistência técnica.*

R. Luis de Camões, 35-B 1300 Lisboa Tel:363 78 64 Fax-641910
R. Joaquim Paço D'Arcos, 9-A 1500 Lisboa Tel: 715 3159 Fax-7150770

Não necessita recortar qualquer Cupão !!!

Bastará uma fotocópia ou, simplesmente, uma carta com os elementos solicitados !

CUPÃO DE ASSINATURA

Assinaturas: Continente e Ilhas - 6 n^os = 1.800\$00 ; 11 n^os = 3.000\$00

Estrangeiro - 6 n^os = 3.600\$00 ; 11 n^os = 6.000\$00

Desejo assinar a revista "RS232-Informática" por um período correspondente a: _____ números. A iniciar no N^o _____

Nome _____

Morada _____

Localidade _____ Cód.Postal _____

Junto envio Cheque N^o _____ Vale Postal N^o _____

CUPÃO DE PEDIDOS

Em relação a software, só serão satisfeitos os pedidos cujos títulos constem das listas.

No que respeita à Secção "Biblioteca", poderá acontecer que alguns dos títulos se encontrem "Esgotados" face ao número limitado de Importação e haja alteração de preços sem aviso prévio.

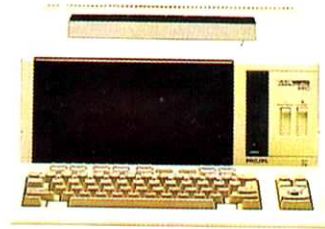
Nome _____

Morada _____

Localidade _____ Cód. Postal _____

Designação	Título	Código	Valor
COMPUTADOR _____		Valor	\$
		Portes	200\$00
JUNTO ENVIO CHEQUE N ^o _____ VALE POSTAL N ^o _____		Total	\$

NOTA: Em virtude de um número significativo de encomendas, enviadas "À cobrança", não terem sido levantadas pelos seus destinatários, com prejuízo para "RS232", anulámos esta modalidade. No caso de não nos ser possível satisfazer a totalidade do seu pedido, remeteremos a a quantia em excesso.



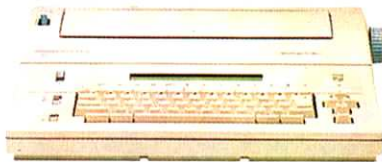
CHAI
INFORMÁTICA



CHAI 01
C.C.S.JOÃO DE DEUS - Lj 428 **LOJA**
Tel: 01 - 77 94 52 LISBOA



CHAI 02
R. DA MADALENA, 138 / 144 **ARMAZÉM**
Tel: 01 - 86 64 41 LISBOA



CHAI 03
R.DA MADALENA, 124 **SEDE**
Tel: 01 - 86 39 50



RDS - NETBIT
C.C.OLAIAS - Lj 103 **PROJECTOS**
Tel: 01 - 89 55 93



► **Tecnologias de Informação**
Compatíveis **Consigo**

DISTRIBUIDOR
AUTORIZADO





Você não é Hemingway

Hemingway é um dos vultos da cultura universal que melhor exprime a ideia de *poder pessoal*. Hemingway é o triunfo do indivíduo, oposto ao princípio, hoje decadente, da superioridade das massas. Esta ideia do *poder pessoal* caracteriza, desde o primeiro momento, toda a filosofia da Apple. O computador, para a Apple, é uma faculdade do indivíduo, que serve, não para uniformizá-lo, mas para torná-lo singular. Cada utilizador do Macintosh da Apple descobre-se diferente de todos os outros homens — mesmo dos que exercem idêntica actividade — porque descobre, em si mesmo, capacidades únicas que a intimidade com o computador lhe revela. Qualquer que seja a sua actividade, a Apple propõe-lhe o maior de todos os conhecimentos: *conheça-se a si mesmo*.



A força de ser melhor!